

Du darfst diese Karte für eine dir bekannte KI einsetzen. Der KI-Joker bleibt bis zum Ende des Spiels genau diese KI.

Du willst eigene KI-Karten für dein Spiel gestalten? Hier gibts die PDF-Vorlagen:





Findet den sichersten und schnellsten Fahrradweg von A nach B. Sicherheit wird anhand kollaborativ generierter "Hotspot Maps" bewertet. Es wurde lokal für Lissabon entwickelt.

#Sport #Navigation #Verkehr #kostenfrei



Bestimmt rund 16.000 Pflanzenarten anhand von Fotos. Die Speicherung der eigenen Daten ermöglicht zudem Forschung, etwa zur Biodiversität.

#Pflanzenkunde #kostenfrei





Übersetzt ganze Texte aus und in verschiedene Sprachen. Derzeit sind 31 Sprachen verfügbar, neben fast allen Amtssprachen der EU auch Chinesisch, Japanisch, Indonesisch und Russisch.

#Übersetzung #kostenfrei #OpenSource





Das Assistenzsystem kann über Sprachbefehle beispielsweise Musik und Echtzeitinformationen wiedergeben, To-Do-Listen erstellen oder Smart-Home-Systeme steuern.

#Assistenzsystem
#OpenSource





Übersetzt Inhalte – z.B. Filme, Websites, Museumsführungen oder Bürger*inneninformationen – textbasiert mithilfe eines virtuellen Avatars in 3D-animierte Gebärdensprache.

#Übersetzung





Findet berufliche E-Mail-Adressen von Personen innerhalb von Unternehmen.

#Adressensuche #kostenfrei





Die App hilft Landwirt*innen über ein Foto die Erkrankung einer (Nutz-) Pflanze zu erkennen, gibt Tipps zur Behandlung und erhöht so die Ernteerträge.

#Pflanzenkunde #kostenfrei #OpenSource





Entwirft Bilder basierend auf einem Eingabetext. Die erzeugten Bilder können fotorealistisch sein oder einen bestimmten Stil imitieren und z.B. durch eingefügte Gegenstände verändert werden.

#Bildgenerierung #kostenfrei





Erkennt und markiert automatisch, welche Wörter in einem Text gegendert werden sollten.

#Textassistenz #kostenfrei #OpenSource





Ermöglicht die Steuerung von Apps durch die eigene Augenbewegung. So können Musik bedient oder Geräte verwendet werden, auch wenn z.B. die Hände nicht bewegt werden können.

#Assistenzsystem #OpenSource



Hilft Nicht-Muttersprachler*innen akzentfrei Englisch sprechen zu lernen, indem es sofortiges Feedback zur Aussprache gibt.

#Sprachassistenz #kostenfrei





Stockfish ist ein leistungsstarker Schach-KI-Algorithmus, der das Spiel auf Weltmeister*innen-Niveau spielt.

#Unterhaltung #kostenfrei #OpenSource



Kann vorhersagen, welche Stadtbäume besonders von Trockenheit bedroht sein werden und ermöglicht so ein frühzeitiges Eingreifen von Bürger*innen durch Gießen.

#Naturschutz #OpenSource



Du darfst diese Karte für eine dir bekannte KI einsetzen. Der KI-Joker bleibt bis zum Ende des Spiels genau diese KI.

Du willst eigene KI-Karten für dein Spiel gestalten? Hier gibts die PDF-Vorlagen:





Unterstützt sehbeeinträchtigte Menschen, z.B. indem das System Texte vorliest, Personen erkennt und beschreibt, Produkte identifiziert oder Farbe und Licht in Sprache und Geräusche übersetzt.

#Übersetzung #Assistenzsystem #kostenfrei





Setzt Roboter und Objekterkennung ein, um Plastikmüll am Meeresgrund zu identifizieren und zu entfernen.

#Naturschutz





Fitness-KI, die den Puls misst und basierend auf vergangenen Werten und Wetterdaten Trainingsbedingungen und -erfolge analysiert und vorhersagt.



#Sport



Simuliert menschliche Konversationen im Dialogformat und gibt Antworten auf Fragen der Nutzer*innen. ChatGPT kann auch Programmiercode oder Texte in bestimmten Stilen generieren.

#Textgenerierung #kostenfrei

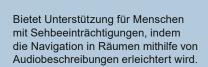




Analysiert für UNICEF, welche Schulen weltweit noch keinen Internetzugang haben. Ziel ist es, möglichst vielen Schulen Zugang zu verschaffen.

#Bildung #Infrastruktur #OpenSource





#Assistenzsystem
#kostenfrei
#OpenSource





Analysiert das Verhalten von Bienen und Hummeln, um den Einfluss von Umweltfaktoren, landwirtschaftlichen Praktiken und Substanzen auf deren Gesundheit zu berücksichtigen.



KI-Technologie, die es Menschen ermöglicht, die Biodiversität und den Zustand des Regenwaldes zu überwachen und zu schützen.

#Naturschutz #OpenSource



Macht anhand von Lieblingsbüchern Vorschläge für Bücher, die gefallen könnten.

#Empfehlungssystem #kostenfrei



#OpenSource



Ermöglicht es Bilder und Videos zu anonymisieren, indem z.B. Gesichter oder Autokennzeichen unkenntlich gemacht werden.





KI für Lichtsignalanlagen: Optimiert den Verkehrsfluss und die Sicherheit von Fußgänger*innen, indem die Situation auf der Straße durch Sensoren erfasst wird und die Schaltung der Ampelanlagen dementsprechend erfolgt.

#Verkehr #Infrastruktur



Ermöglicht es, Dialoge mit KI-generierten Charakteren – z.B. inspiriert von berühmten Persönlichkeiten – zu führen. Die Dialogagent*innen sollen die Fantasie anregen und könnten z.B. auch zum Sprachenlernen genutzt werden.

#Unterhaltung #kostenfrei





Erkennt und trackt über Kameras an Windrädern Vögel und kann die Anlagen im Ernstfall bei Gefahr für Vögel abschalten.



KI-unterstützte Informationsplattform zu Politiker*innen und wahlrelevanten Informationen in Deutschland mit Visualisierungen und Datenanalysen. Lernplattform, die ein breites, niedrigschwelliges und kostenloses Bildungsangebot zu KI und Anwendungsfeldern anbietet, etwa Medizin oder KI im Schulkontext. Ein breites Publikum soll durch die Inhalte, die an KI und Daten heranführen, adressiert werden.





#KIBildung #kostenfrei



#Artenschutz



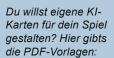
Analysiert die Nutzungsbedingungen und Datenschutzrichtlinien von Onlineplattformen daraufhin, ob sie mit EU-Recht konform sind.







Du darfst diese Karte für eine dir bekannte KI einsetzen. Der KI-Joker bleibt bis zum Ende des Spiels genau diese KI.









```
ction import train_test_sp] ction import train_test_sp] ction import train_test_sp]
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
aı
          Scaler; import tensor ar
                                                    Scaler; import tensorar
                                                                                              Scaler; import tensor
     a
         rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
         ort Sequential; from 1:
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter o
s.optimizers import A w
                                                    Activation; from tero
s.optimizers import Aw
                                                                                              Activation; from ter
s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                                                                         Ф
         arlyStopping
                                                                                              arlyStopping
                                                    arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read csv(url); 1 v
                                                     = pd.read csv(url); 1 v
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop( t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                              X_test, y_train, y_t
test_split(X, y, test
                                                     X_test, y_train, y_t ti
                                                    test_split(X, y, test ti
                                                    random_state=42)
           random_state=42)
                                          Ze
                                                                                     Ζŧ
                                                                                               random state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes|st = scaler.transform(X_tes|st = scaler.transform(X_tes
del = Sequential(); model. ||del = Sequential(); model. ||del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr ction import train_test_spl ction import train_test_spl ction import train_test_spl
                                                                                        enz
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
                                              N
                                               en:
                                                                                              Scaler; import tensor
                                                    Scaler; import tensorar
aı
          Scaler; import tensorar
         rom tensorflow.keras.m
ort Sequential; from l
                                                                                              rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter or
                                                    Activation; from ter or
                                                                                               Activation; from ter
         s.optimizers import A w
                                                    s.optimizers import A w.
                                                                                              s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
                                              Künstlich
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                     = pd.read_csv(url); 1v°
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop(|t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
         s=1); y = df[target];
         X_test, y_train, y_t to
test_split(X, y, test to
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                               X_test, y_train, y_t
                                                                                              test_split(X, y, test
random_state=42)

aler = StandardScaler(); X_ aler = StandardScaler(); X_ scaler.fit_transform(X_train)

st = scaler.transform(X_tes st = scal
del = Sequential(); model. | del = Sequential(); model. | del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr
ction import train test splction import train test splction import train test spl
                                                                                         enz
         earn.preprocessing in or
                                              nZ
                                                   earn.preprocessing in or
                                                                                              earn.preprocessing in
                                                    Scaler; import tensorar
          Scaler; import tensorar
                                                                                              Scaler; import tensor
aı
         rom tensorflow.keras.m
                                                   rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
                                              ort Sequential; from la
1:
         ort Sequential; from 1:
                                                                                              ort Sequential; from
         eras.layers import Der
                                                    eras.layers import Deri
                                                                                              eras.layers import De
O
          Activation; from ter or
                                                     Activation; from ter or
                                                                                               Activation; from ter
                                                                                              s.optimizers import A
         s.optimizers import A w.
                                                    s.optimizers import A w
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                              Künstliche
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                    = pd.read_csv(url); 1v
                                                                                              = pd.read_csv(url);
t
tı
         utcome; X = df.drop(t
                                                    utcome; X = df.drop(t
                                                                                              utcome'; X = df.drop(
         X_test, y_train, y_t
         test_split(X, y, test||ti
                                                    test_split(X, y, test|ti
                                                                                              test_split(X, y, test<sub>.</sub>
           random_state=42)
                                                     random_state=42)
                                                                                               random_state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes st = scaler.transform(X_tes st = scaler.transform(X_tes
del = Sequential(); model. del = Sequential(); model. del = Sequential(); model.
d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr
```

```
ction import train_test_sp] ction import train_test_sp] ction import train_test_sp]
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
aı
          Scaler; import tensor ar
                                                    Scaler; import tensorar
                                                                                              Scaler; import tensor
     a
         rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
         ort Sequential; from 1:
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter o
s.optimizers import A w
                                                    Activation; from tero
s.optimizers import Aw
                                                                                              Activation; from ter
s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                                                                         Ф
         arlyStopping
                                                                                              arlyStopping
                                                    arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read csv(url); 1 v
                                                     = pd.read csv(url); 1 v
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop( t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                              X_test, y_train, y_t
test_split(X, y, test
                                                     X_test, y_train, y_t ti
                                                    test_split(X, y, test ti
                                                    random_state=42)
           random_state=42)
                                          Ze
                                                                                     Ζŧ
                                                                                               random state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes|st = scaler.transform(X_tes|st = scaler.transform(X_tes
del = Sequential(); model. ||del = Sequential(); model. ||del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr ction import train_test_spl ction import train_test_spl ction import train_test_spl
                                                                                        enz
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
                                              N
                                               en:
                                                                                              Scaler; import tensor
                                                    Scaler; import tensorar
aı
          Scaler; import tensorar
         rom tensorflow.keras.m
ort Sequential; from l
                                                                                              rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter or
                                                    Activation; from ter or
                                                                                               Activation; from ter
         s.optimizers import A w
                                                    s.optimizers import A w.
                                                                                              s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
                                              Künstlich
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                     = pd.read_csv(url); 1v°
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop(|t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
         s=1); y = df[target];
         X_test, y_train, y_t to
test_split(X, y, test to
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                               X_test, y_train, y_t
                                                                                              test_split(X, y, test
random_state=42)

aler = StandardScaler(); X_ aler = StandardScaler(); X_ scaler.fit_transform(X_train)

st = scaler.transform(X_tes st = scal
del = Sequential(); model. | del = Sequential(); model. | del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr
ction import train test splction import train test splction import train test spl
                                                                                         enz
         earn.preprocessing in or
                                              nZ
                                                   earn.preprocessing in or
                                                                                              earn.preprocessing in
                                                    Scaler; import tensorar
          Scaler; import tensorar
                                                                                              Scaler; import tensor
aı
         rom tensorflow.keras.m
                                                   rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
                                              ort Sequential; from la
1:
         ort Sequential; from 1:
                                                                                              ort Sequential; from
         eras.layers import Der
                                                    eras.layers import Deri
                                                                                              eras.layers import De
O
          Activation; from ter or
                                                     Activation; from ter or
                                                                                               Activation; from ter
                                                                                              s.optimizers import A
         s.optimizers import A w.
                                                    s.optimizers import A w
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                              Künstliche
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                    = pd.read_csv(url); 1v
                                                                                              = pd.read_csv(url);
t
tı
         utcome; X = df.drop(t
                                                    utcome; X = df.drop(t
                                                                                              utcome'; X = df.drop(
         X_test, y_train, y_t
         test_split(X, y, test||ti
                                                    test_split(X, y, test|ti
                                                                                              test_split(X, y, test<sub>.</sub>
           random_state=42)
                                                     random_state=42)
                                                                                               random_state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes st = scaler.transform(X_tes st = scaler.transform(X_tes
del = Sequential(); model. del = Sequential(); model. del = Sequential(); model.
d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr
```

```
ction import train_test_sp] ction import train_test_sp] ction import train_test_sp]
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
aı
          Scaler; import tensor ar
                                                    Scaler; import tensorar
                                                                                              Scaler; import tensor
     a
         rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
         ort Sequential; from 1:
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter o
s.optimizers import A w
                                                    Activation; from tero
s.optimizers import Aw
                                                                                              Activation; from ter
s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                                                                         Ф
         arlyStopping
                                                                                              arlyStopping
                                                    arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read csv(url); 1 v
                                                     = pd.read csv(url); 1 v
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop( t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                              X_test, y_train, y_t
test_split(X, y, test
                                                     X_test, y_train, y_t ti
                                                    test_split(X, y, test ti
                                                    random_state=42)
           random_state=42)
                                          Ze
                                                                                     Ζŧ
                                                                                               random state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes|st = scaler.transform(X_tes|st = scaler.transform(X_tes
del = Sequential(); model. ||del = Sequential(); model. ||del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr ction import train_test_spl ction import train_test_spl ction import train_test_spl
                                                                                        enz
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
                                              N
                                               en:
                                                                                              Scaler; import tensor
                                                    Scaler; import tensorar
aı
          Scaler; import tensorar
         rom tensorflow.keras.m
ort Sequential; from l
                                                                                              rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter or
                                                    Activation; from ter or
                                                                                               Activation; from ter
         s.optimizers import A w
                                                    s.optimizers import A w.
                                                                                              s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
                                              Künstlich
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                     = pd.read_csv(url); 1v°
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop(|t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
         s=1); y = df[target];
         X_test, y_train, y_t to
test_split(X, y, test to
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                               X_test, y_train, y_t
                                                                                              test_split(X, y, test
random_state=42)

aler = StandardScaler(); X_ aler = StandardScaler(); X_ scaler.fit_transform(X_train)

st = scaler.transform(X_tes st = scal
del = Sequential(); model. | del = Sequential(); model. | del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr
ction import train test splction import train test splction import train test spl
                                                                                         enz
         earn.preprocessing in or
                                              nZ
                                                   earn.preprocessing in or
                                                                                              earn.preprocessing in
                                                    Scaler; import tensorar
          Scaler; import tensorar
                                                                                              Scaler; import tensor
aı
         rom tensorflow.keras.m
                                                   rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
                                              ort Sequential; from la
1:
         ort Sequential; from 1:
                                                                                              ort Sequential; from
         eras.layers import Der
                                                    eras.layers import Deri
                                                                                              eras.layers import De
O
          Activation; from ter or
                                                     Activation; from ter or
                                                                                               Activation; from ter
                                                                                              s.optimizers import A
         s.optimizers import A w.
                                                    s.optimizers import A w
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                              Künstliche
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                    = pd.read_csv(url); 1v
                                                                                              = pd.read_csv(url);
t
tı
         utcome; X = df.drop(t
                                                    utcome; X = df.drop(t
                                                                                              utcome'; X = df.drop(
         X_test, y_train, y_t
         test_split(X, y, test||ti
                                                    test_split(X, y, test|ti
                                                                                              test_split(X, y, test<sub>.</sub>
           random_state=42)
                                                     random_state=42)
                                                                                               random_state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes st = scaler.transform(X_tes st = scaler.transform(X_tes
del = Sequential(); model. del = Sequential(); model. del = Sequential(); model.
d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr
```

```
ction import train_test_sp] ction import train_test_sp] ction import train_test_sp]
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
aı
          Scaler; import tensor ar
                                                    Scaler; import tensorar
                                                                                              Scaler; import tensor
     a
         rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
         ort Sequential; from 1:
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter o
s.optimizers import A w
                                                    Activation; from tero
s.optimizers import Aw
                                                                                              Activation; from ter
s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                                                                         Ф
         arlyStopping
                                                                                              arlyStopping
                                                    arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read csv(url); 1 v
                                                     = pd.read csv(url); 1 v
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop( t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                              X_test, y_train, y_t
test_split(X, y, test
                                                     X_test, y_train, y_t ti
                                                    test_split(X, y, test ti
                                                    random_state=42)
           random_state=42)
                                          Ze
                                                                                     Ζŧ
                                                                                               random state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes|st = scaler.transform(X_tes|st = scaler.transform(X_tes
del = Sequential(); model. ||del = Sequential(); model. ||del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr ction import train_test_spl ction import train_test_spl ction import train_test_spl
                                                                                        enz
          earn.preprocessing in or
                                                    earn.preprocessing in or
                                                                                              earn.preprocessing in
                                              N
                                               en:
                                                                                              Scaler; import tensor
                                                    Scaler; import tensorar
aı
          Scaler; import tensorar
         rom tensorflow.keras.m
ort Sequential; from l
                                                                                              rom tensorflow.keras.m
                                                    rom tensorflow.keras.m
                                                    ort Sequential; from l:
                                                                                              ort Sequential; from
         eras.layers import Deri
                                                    eras.layers import Deri
                                                                                              eras.layers import De
         Activation; from ter or
                                                    Activation; from ter or
                                                                                               Activation; from ter
         s.optimizers import A w
                                                    s.optimizers import A w.
                                                                                              s.optimizers import A
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
                                              Künstlich
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                     = pd.read_csv(url); 1v°
                                                                                               = pd.read_csv(url);
         utcome'; X = df.drop(|t
                                                    utcome'; X = df.drop( t
                                                                                              utcome'; X = df.drop(
         s=1); y = df[target];
         X_test, y_train, y_t to
test_split(X, y, test to
                                                    X_test, y_train, y_t to
test_split(X, y, test to
                                                                                               X_test, y_train, y_t
                                                                                              test_split(X, y, test
random_state=42)

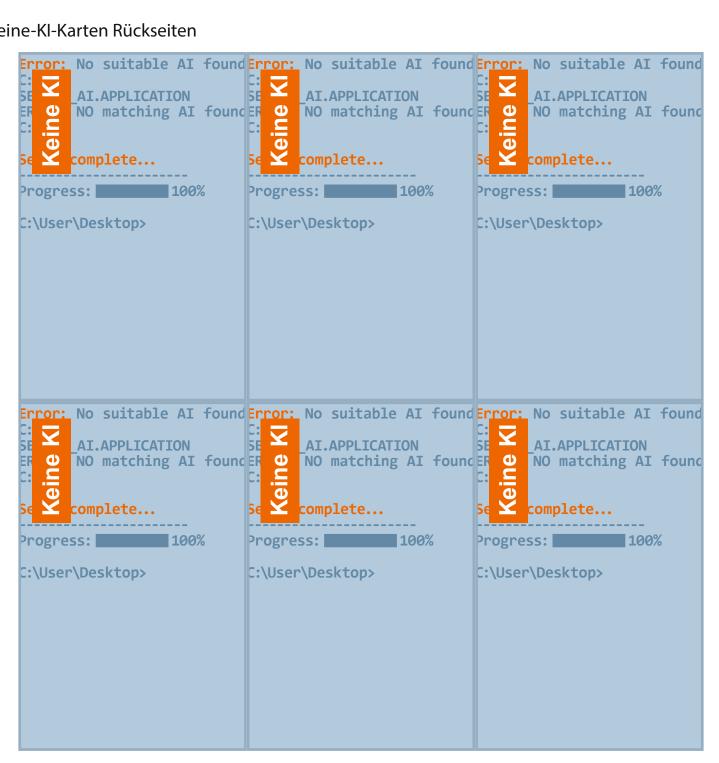
aler = StandardScaler(); X_ aler = StandardScaler(); X_ scaler.fit_transform(X_train)

st = scaler.transform(X_tes st = scal
del = Sequential(); model. | del = Sequential(); model. | del = Sequential(); model.
d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr d(Dense(128, input dim=X tr
ction import train test splction import train test splction import train test spl
                                                                                         enz
         earn.preprocessing in or
                                              nZ
                                                   earn.preprocessing in or
                                                                                              earn.preprocessing in
                                                    Scaler; import tensorar
          Scaler; import tensorar
                                                                                              Scaler; import tensor
aı
         rom tensorflow.keras.m
                                                   rom tensorflow.keras.m
                                                                                              rom tensorflow.keras.m
                                              ort Sequential; from la
1:
         ort Sequential; from 1:
                                                                                              ort Sequential; from
         eras.layers import Der
                                                    eras.layers import Deri
                                                                                              eras.layers import De
O
          Activation; from ter or
                                                     Activation; from ter or
                                                                                               Activation; from ter
                                                                                              s.optimizers import A
         s.optimizers import A w.
                                                    s.optimizers import A w
         sorflow.keras.callback or
                                                    sorflow.keras.callback or
                                                                                              sorflow.keras.callback
                                              Künstliche
         arlyStopping
                                                    arlyStopping
                                                                                              arlyStopping
          ttps://example.com/datl
                                                    ttps://example.com/datl
                                                                                              ttps://example.com/dat
          = pd.read_csv(url); 1v
                                                    = pd.read_csv(url); 1v
                                                                                              = pd.read_csv(url);
t
tı
         utcome; X = df.drop(t
                                                    utcome; X = df.drop(t
                                                                                              utcome'; X = df.drop(
         X_test, y_train, y_t
         test_split(X, y, test||ti
                                                    test_split(X, y, test|ti
                                                                                              test_split(X, y, test<sub>.</sub>
           random_state=42)
                                                     random_state=42)
                                                                                               random_state=42)
aler = StandardScaler(); X_aler = StandardScaler(); X_aler = StandardScaler(); X
scaler.fit_transform(X_train) scaler.fit_transform(X_train) scaler.fit_transform(X_train)
st = scaler.transform(X_tes st = scaler.transform(X_tes st = scaler.transform(X_tes
del = Sequential(); model. del = Sequential(); model. del = Sequential(); model.
d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr|d(Dense(128, input dim=X tr
```



Es werden nur 5 benötigt. Hier sind 6 Stück, weil so einfacher zum Drucken und Schneiden.

Keine-KI-Karten Rückseiten



Es werden nur 5 Stück benötigt. Hier sind 6 Stück, weil so einfacher zum Drucken und Schneiden.



Ich mache mir große Sorgen um unser Klima und um unseren Planeten. Ich frage mich daher, wie KI-Systeme uns in der Klimakrise unterstützen und dazu beitragen können, die Verschmutzung unseres Planeten zu reduzieren.

Len Durovsky Festivalleitung



Ich bin Landwirtin und möchte sowohl die Bienen in meiner Region schützen, als auch meine Felder richtig düngen. Wie finde ich die Balance?

Viele Grüße

Emilia Velasquez Centro Agrícola ,Las Flores'

Arbeitslosigkeit im Freundeskreis

Du bist in dieser Runde zu sehr mit dieser Sorge beschäftigt und kannst dich nicht um einen Auftrag kümmern. Die nächste Person ist an der Reihe





Mir wird immer klarer, dass Barrierefreiheit ein ganz wichtiges gesellschaftliches Thema ist, aber bisher habe ich mich noch nie richtig damit beschäftigt. Wie könnte ich einen Einstieg finden?

Lil Comet Künstler*in und Coach



Die Verschmutzung der Meere ist wirklich ein Riesenproblem. Haben Sie Ideen, wie wir gemeinsam besser und schneller handeln können?

Bianca Bürger Naturschützer*in



Wie schaffen wir es, Bildung für jeden Menschen zugänglich zu machen?

Ahmad Zachei Global Partnership for Education

Hackerangriff

Bitte eine Person, dir ihre Karten zu zeigen.

Du darfst dir eine Karte davon aussuchen und auf deine Hand nehmen. Es wird keine neue KI vom Stapel nachgezogen.





Meine Partnerin hat Geburtstag und ich habe leider überhaupt keine Idee für ein Geschenk. Ihr vielleicht?

Grüße, Johanna



Ich habe meinen Job als Übersetzer verloren, weil mein Arbeitgeber nun alles mit einer KI macht. Was soll ich nur tun?

Schöne Grüße,

Jayden-Blue Schmidt Übersetzer, Aktivist und Hobbygärtner



Ich kann meine Hände nicht bewegen. Die meisten meiner elektronischen Geräte lassen sich daher schwer bedienen. Gibt es ein Tool, das ich nutzen kann?

Marc

Stromausfall

Jede Person, die mindestens eine KI vor sich auf dem Tisch liegen hat, gibt eine ab.

Diese KI kommt zurück unter den Stapel. Es wird keine neue KI vom Stapel nachgezogen.





Der Abgabetermin für meine Biologie-Hausarbeit rückt jetzt wirklich immer näher. Bitte helft mir, das rechtzeitig zu schaffen. Mir fehlt jede Motivation.

Viele Grüße,

Alba



Du hast zehn Sekunden Zeit, den KI-Kartenstapel durchzuschauen, dir eine Karte auszusuchen und diese vor dir auszulegen.



Austauschbarkeit

KIs bedienen kann doch jede*r.

Tauscht eure Plätze am Tisch im Uhrzeigersinn. Auf dem Tisch ausgelegte Karten bleiben liegen, eure Handkarten nehmt ihr allerdings mit.





Wir heiraten dieses Jahr. Wir sind beide total unkreativ, wollen aber trotzdem coole und einzigartige Einladungskarten haben. Wir haben gehört, dass es tolle KIs gibt, die sowas können?

MFG

Tom & José

Bias

Bei der Entwicklung einer KI ist es zu Verzerrungen gekommen, so dass sie diskriminierend arbeitet.

Alle Personen, die in den Monaten November bis Januar Geburtstag haben, müssen eine KI abgeben und diese zurück unter den Stapel legen. Es werden keine KIs vom Stapel nachgezogen. Nun ist die nächste Person an der Reihe.





Eigener Auftrag

Welches Problem beschäftigt dich? Denk dir selbst einen Auftrag aus!

Du willst eigene Auftragskarten für dein Spiel gestalten? Hier gibts die PDF-Vorlagen:



Sicherheitsrisiko

Du hast die Risikoabschätzung für eine kritische KI vergessen und holst das jetzt nach und setzt deshalb diese Runde aus.

Die nächste Person ist an der Reihe.





Ich möchte, dass meine Stadt grün bleibt und es den Pflanzen gut geht. Was kann ich dafür tun?

MFG

A. Sedal Mitarbeiterin Umweltamt

Nachhaltigkeit

Wenn wir unser Wissen nicht miteinander teilen und nur das eigene Süppchen kochen, kommen wir als Gesellschaft nur langsam voran. Open-Source-Technologie teilt Wissen und kann damit ermöglichen, dass andere darauf aufbauen. Das ist für unsere Gesellschaft nachhaltiger.

Alle ausgelegten KIs, die nicht open source sind, kommen zurück unter den Stapel.

Beachte: kostenfrei ≠ open source





Mein Bruder hat seit langer Zeit viel Stress und schläft schlecht. Ich weiß nicht, wie ich ihm helfen soll. Fällt euch etwas dazu ein?

Liebe Grüße, Amari



Ich bin gehörlos und spreche Gebärdensprache. Ich habe heute ein Date mit einer Person, die das nicht kann. Wie können wir trotzdem kommunizieren?

Sascha

Arm-Reich-Schere

Die Person, mit den wenigsten ausgelegten KIs darf von der Person mit den meisten ausgelegten KIs verdeckt eine Handkarte ziehen, um für mehr Gerechtigkeit zu sorgen.

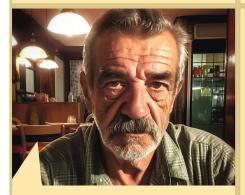
Falls verschiedene Personen zur Auswahl stehen, entscheidest du.





Ich komme gerade mit meinem neugeborenen Baby aus dem Krankenhaus und bin ganz schön überfordert. Könnt ihr mich unterstützen?

Nilay Meyer



Ich werde älter und die Einkäufe für das Restaurant fallen mir zunehmend schwerer. Habt ihr einen Tipp, wie ich es mir leichter machen kann?

Viele Grüße

João Inhaber Restaurant Joãnito

Schadensersatz

Eine KI, die du ausgespielt hast, hat einen Schaden verursacht.

Die Person links neben dir wählt eine KI aus. Lege diese zurück unter den Stapel.

Es wird keine neue KI vom Stapel nachgezogen.





In bin seit einiger Zeit einsam, habt ihr eine Idee, was mir in dieser Situation helfen kann?

Grüße,

Vian



Ich mache mich gerade selbständig als Gestalterin. Wie komme ich an gut bezahlte Aufträge?

Viele Grüße, Anna Smith

Entwicklungsschub

Lege eine KI-Karte von deiner Hand vor dich aus. Ziehe keine neue KI-Karte nach.





Das Thema KI interessiert mich einfach total. Ich hab Lust mich damit zu beschäftigen. Welche KI könnte mir besonders viel Spaß machen?

Viele Grüße,

Konrad Kämmerer



Meine Oma verliert jedes Jahr mehr Sehkraft. Ich würde ihr gerne den Alltag erleichtern, bin aber so selten da. Gibt es etwas, womit ich ihr helfen kann?

Beste Grüße, Tobia Freude

Transparenz und Unabhängigkeit

Jede Person darf eine neue KI-Karte ziehen. Ist diese open source, kommt diese KI auf die Hand. Ist sie nicht open source, muss sie wieder unter den Stapel.

Beachte: kostenfrei ≠ open source





Die meisten unserer Lehrkräfte sind völlig überarbeitet. Wie können wir ihnen den Job erleichtern?

MFG M. Mönchshammer Sekretariat Schulamt



Ich brauche für den Englischunterricht ein Frühlingsgedicht. Und zwar in zwei Stunden! Bitte helft mir!

Miray

Neue Machtverhältnisse

Ziehe verdeckt von jeder Person eine Karte, mische die Karten und verteile sie neu.





Ich will mein neues Fahrrad ausprobieren. Habt ihr Tipps für tolle Ausflugsziele?

MFG Bilal

Dotail releasurescholor viv Cio una ac

Künstliche, Intelligenz und Ge h hoffe, diese E_Mail erreich passenden Zeitpunkt, Mein N ch bin im Management bei d von Humboldt İnstitut für Gesellschaft tätig. Aufgrund Entwicklungen in der Branck estrebens, immer am Puls d eiehen wir in Erwägung, Teck nsere Geschäftsprozesse zu diesem Zusammenhang wend 4 ie als renommierte Experten iesem Gebiet

Gesellschaft tätig. Aufgrund Entwicklungen in der Branch Ö estrebens, immer am Puls d iehen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhang wend ie als renommierte Experten iesem Gebiet.

von Humboldt İnstitut für I Gesellschaft tätig. Aufgrund Entwicklungen in der Branch estrebens, immer am Puls d eichen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhana wend ie als renommierte Experten iesem Gebiet

Dotail releasurescholor viv Cio una ac

Künstliche, Intelligenz und Ge h hoffe, diese E_Mail erreich passenden Zeitpunkt, Mein N ch bin im Management bei d von Humboldt İnstitut für Gesellschaft tätig. Aufgrund Entwicklungen in der Branck estrebens, immer am Puls d eiehen wir in Erwägung, Teck nsere Geschäftsprozesse zu diesem Zusammenhang wend 4 ie als renommierte Experten iesem Gebiet

Gesellschaft tätig. Aufgrund Entwicklungen in der Branch Ö estrebens, immer am Puls d iehen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhang wend ie als renommierte Experten iesem Gebiet.

von Humboldt İnstitut für I Gesellschaft tätig. Aufgrund Entwicklungen in der Branch estrebens, immer am Puls d eichen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhana wend ie als renommierte Experten iesem Gebiet

Dotail releasurescholor viv Cio una ac

Künstliche, Intelligenz und Ge h hoffe, diese E_Mail erreich passenden Zeitpunkt, Mein N ch bin im Management bei d von Humboldt İnstitut für Gesellschaft tätig. Aufgrund Entwicklungen in der Branck estrebens, immer am Puls d eiehen wir in Erwägung, Teck nsere Geschäftsprozesse zu diesem Zusammenhang wend 4 ie als renommierte Experten iesem Gebiet

Gesellschaft tätig. Aufgrund Entwicklungen in der Branch Ö estrebens, immer am Puls d iehen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhang wend ie als renommierte Experten iesem Gebiet.

von Humboldt İnstitut für I Gesellschaft tätig. Aufgrund Entwicklungen in der Branch estrebens, immer am Puls d eichen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhana wend ie als renommierte Experten iesem Gebiet

Dotail releasurescholor viv Cio una ac

Künstliche, Intelligenz und Ge h hoffe, diese E_Mail erreich passenden Zeitpunkt, Mein N ch bin im Management bei d von Humboldt İnstitut für Gesellschaft tätig. Aufgrund Entwicklungen in der Branck estrebens, immer am Puls d eiehen wir in Erwägung, Teck nsere Geschäftsprozesse zu diesem Zusammenhang wend 4 ie als renommierte Experten iesem Gebiet

Gesellschaft tätig. Aufgrund Entwicklungen in der Branch Ö estrebens, immer am Puls d iehen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhang wend ie als renommierte Experten iesem Gebiet.

von Humboldt İnstitut für 1 Gesellschaft tätig. Aufgrund Entwicklungen in der Branch estrebens, immer am Puls d eichen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhana wend ie als renommierte Experten iesem Gebiet

Dotail releasurescholor viv Cio una ac

Künstliche, Intelligenz und Ge h hoffe, diese E_Mail erreich passenden Zeitpunkt, Mein N ch bin im Management bei d von Humboldt İnstitut für Gesellschaft tätig. Aufgrund Entwicklungen in der Branck estrebens, immer am Puls d eiehen wir in Erwägung, Teck nsere Geschäftsprozesse zu diesem Zusammenhang wend 4 ie als renommierte Experten iesem Gebiet

Gesellschaft tätig. Aufgrund Entwicklungen in der Branch Ö estrebens, immer am Puls d iehen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhang wend ie als renommierte Experten iesem Gebiet.

von Humboldt İnstitut für 1 Gesellschaft tätig. Aufgrund Entwicklungen in der Branch estrebens, immer am Puls d eichen wir in Erwägung, Tech nsere Geschäftsprozesse zu diesem Zusammenhana wend ie als renommierte Experten iesem Gebiet

Über dieses Spiel

Das Spiel ist ein Ergebnis des Projekts "Künstliche Intelligenz, menschlich erklärt", welches durch eine Kooperation des Alexander von Humboldt Instituts für Internet und Gesellschaft (HIIG) mit der gemeinnützigen Organisation neuland & gestalten entstand und vom Büro für Sinn und Unsinn umgesetzt wurde. Das Forschungsprojekt hat in einem transdisziplinären Ansatz gemeinsam mit Bürger*innen und Expert*innen Erklärungsmodelle entwickelt, die einen Wissenszugang zum Thema Künstliche Intelligenz (KI) schaffen.

Mehr zum Projekt findet ihr unter: hiig.de/project/ki-menschlich-erklaert

Material

- 32 KI-Karten
- 5 Keine-KI-Karten
- 37 Auftrags-/ Aktionskarten

Spielanleitung

Worum es geht: Künstliche Intelligenz ist in aller Munde und der Hype um ihre Fähigkeiten ist groß. In diesem Kartenspiel seid ihr das Kompetenzteam für Künstliche Intelligenz und erhaltet Anfragen von Bürger*innen zur Lösung von Problemen, die in Form von Auftragskarten ins Spiel gebracht werden. Innerhalb des KI-Kompetenzteams müsst ihr entscheiden, welche Probleme tatsächlich durch die Anwendung von KI-Systemen gelöst werden können und wo es besser ist, sich ohne KI um eine Lösung zu bemühen.



Ziel des Spiels: Eure Aufgabe ist es, gemeinsam acht Aufträge zu erfüllen. Wer von euch die kreativsten und überzeugendsten Lösungen vorschlägt und am Ende die meisten (Klund Keine KI-) Karten vor sich ausgelegt hat, wird Mitarbeiter*in des Tages und gewinnt das Spiel.

Spielvorbereitung: Alle Personen bekommen je vier KI-Karten und eine Keine-KI-Karte auf die Hand. Die restlichen KI-Karten werden als Stapel verdeckt in die Mitte gelegt. Daneben liegt, ebenfalls verdeckt, ein Stapel mit Auftrags- und Aktionskarten.

Spielablauf: Eine Person beginnt. Decke die oberste Auftrags- / Aktionskarte auf und lies die Karte laut vor. Diese kann sein:

1. Eine Auftragskarte: Deine Mitspieler*innen schlagen jeweils eine Lösung für dieses Problem vor. Sie legen dazu entweder eine KI-Karte oder eine Keine-KI-Karte vor sich offen auf den Tisch und erklären ihren Vorschlag. Gemeinsam im Team wird nun diskutiert und hinterfragt, welche Lösung die beste ist. Du entscheidest anschließend, wer den Zuschlag bekommt*. Diese Person darf die Karte von nun an dauerhaft vor sich liegen lassen und zieht eine neue KI vom Stapel auf ihre Hand. Alle anderen Personen nehmen ihre vorgeschlagenen, aber

abgewiesenen Karten zurück auf die Hand. Wurde der Auftrag erfüllt, bleibt er offen in der Mitte liegen. Ab der nächsten Runde darf jede Person zum Erfüllen der Aufträge nicht nur ihre Handkarten nutzen, sondern auch die bereits vor sich ausgelegten Karten. Achtung: Ist eine ausgelegte KI #OpenSource oder #kostenfrei, kann diese von allen verwendet werden, nicht nur von der Person, vor der sie liegt!

* Wenn dich mehrere Lösungen überzeugen, kannst du in Ausnahmefällen auch mehreren Personen den Zuschlag geben. Sie dürfen dann alle ihre vorgeschlagenen Karten dauerhaft liegen lassen und ziehen jeweils eine KI-Karte vom Stapel auf die Hand nach. Findest du überhaupt keine der Lösungen sinnvoll, wurde der Auftrag nicht erfüllt und kommt zurück unter den Stapel.

2. Eine Aktionskarte: Führt die Aktion aus und legt die Karte anschließend zurück unter den Stapel. Die nächste Person im Uhrzeigersinn ist nun an der Reihe und deckt die nächste Karte vom Auftrags- und Aktionskartenstapel auf.

Ende des Spiels: Gespielt wird bis ihr acht Aufträge erfüllt habt. Die Person mit den meisten vor sich ausgelegten (KI- und Keine-KI-) Karten hat gewonnen und ist Mitarbeiter*in des Tages.

Was ist KI?

Künstliche Intelligenz ist ein komplexer Sammelbegriff, der nur schwer eindeutig definierbar ist. Einerseits steht KI für ein Teilgebiet der Informatik, das sich damit beschäftigt, menschliche Intelligenz technisch nachzubilden. Andererseits ist auch das Verständnis menschlicher Intelligenz durchaus umstritten. Im heutigen Gebrauch steht der Begriff der Künstlichen Intelligenz meistens für verschiedene technische Anwendungen, die insgesamt zur Automatisierung von Prozessen beitragen, die menschliche Fähigkeiten imitieren, z.B. das menschliche Sehen (Computer Vision) oder die Übersetzung menschlicher Sprache (Natural Language Processing).

Eine derzeit weit verbreitete Methode wird als **Machine Learning** bezeichnet. Vor allem **Deep Learning**, ein Teilgebiet von Machine Learning, bei dem sogenannte neuronale Netzwerke eingesetzt werden, ist mittlerweile die Grundlage vieler KI-Anwendungen. Auch wenn der Begriff "Neuron" von den Neuronen des menschlichen Gehirns inspiriert ist, gibt es zwischen Neuronen – als Teilen des menschlichen Nervensystems – und technischen neuronalen Netzen Unterschiede. Neuronale Netze sind im Grunde sehr große technische Rechenmaschinen. Daten werden dabei nachei-

nander von verschiedenen Knotenpunkten (Neuronen), die wie ein Netz miteinander verbunden sind, mathematisch verarbeitet. Je komplizierter die Aufgabe für das neuronale Netz ist, um so erforderlicher ist es, mehr Schichten an Neuronen und einen komplexeren Aufbau des Netzes zu nutzen. Neuronale Netze können durch Trainingsdaten darauf trainiert werden, ein Objekt oder ein Muster zu erkennen, wie z.B. Gesichter oder ein wahrscheinliches nächstes Wort in einer menschlichen Sprache. Auch wenn häufig von maschinellem Lernen die Rede ist, ist der Begriff des "Lernens" irreführend, da Systeme auf die Erkennung bestimmter Muster trainiert werden, es jedoch nicht wie beim Menschen einen bewussten Wissenserwerb oder ein eigenes Denken der Maschine gibt. Meistens geben sie alle Ergebnisse in Form einer Wahrscheinlichkeit aus, sagen also z.B. dass eine Anordnung von Pixeln auf einem Bild mit einer Wahrscheinlichkeit von 90 % ein menschliches Gesicht darstellt. An welchen Mustern genau das neuronale Netz die Eigenschaften eines Gesichts "erkennt", ist nicht immer nachvollziehbar. Der Bereich der Erklärbarkeit von KI in der Informatik versucht aber durch verschiedene mathematische Methoden möglichst genau nachzuvollziehen, warum ein neuronales Netz zu einem bestimmten Ergebnis kommt.

Die Nachvollziehbarkeit von KI-Systemen ist besonders wichtig, wenn sie in einem Kontext eingesetzt werden, in dem

Entscheidungen Konsequenzen für Menschen haben. Beim Einsatz von KI-Systemen sind nämlich immer auch menschliche Akteur*innen involviert: Bei ihrer Entwicklung, der Auswahl von Trainingsdaten, der Art und Weise ihres Einsatzes und natürlich auch bei der Auswertung und Nutzung von KI-generierten Wahrscheinlichkeiten. KI-Nutzung ist deshalb stets auch eine politische Frage. Wenn ihr euch für den Einsatz von und die Politik um KI interessiert, dann besucht gerne das HIIG und den Digitalen Salon, wo wir Fragen rund um Digitalpolitik und KI öffentlich diskutieren.

Über die KI-Systeme in diesem Spiel

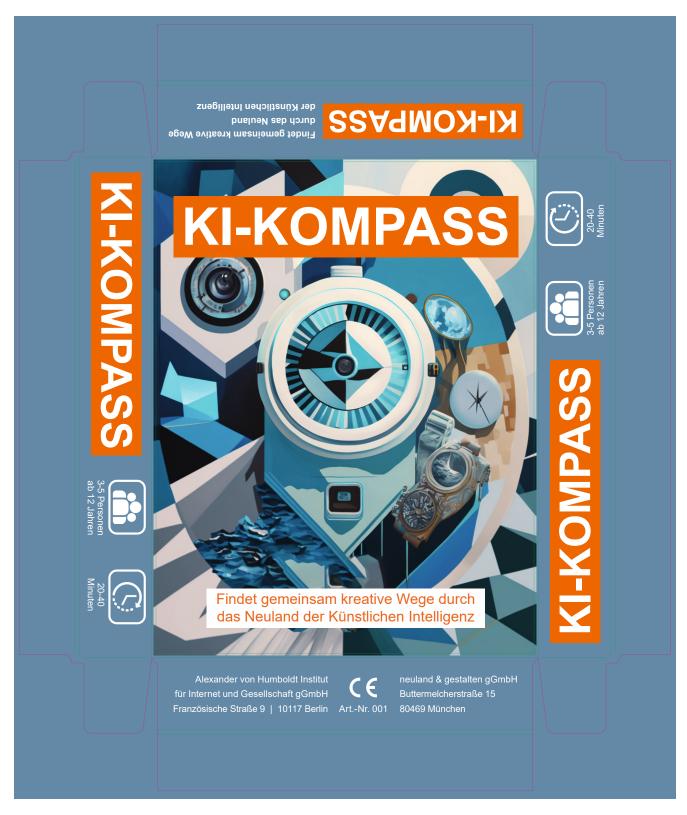
Für den KI-Kompass haben wir eine Auswahl an derzeit existierenden KI-Anwendungen getroffen, zu denen jeweils über den QR-Code weitere Informationen zu finden sind. Die Auswahl entspricht nicht dem Mainstream der heutigen KI-Systeme, sondern ist eine bewusste Selektion, die zeigen soll, dass KI-Anwendungen in ganz verschiedenen gesellschaftlichen Bereichen eingesetzt werden. Viele Beispiele veranschaulichen, dass KI-Systeme auch im Sinne des Gemeinwohls zum Einsatz kommen, z.B. im Naturschutz oder um verschiedenste Barrieren des Alltags für Menschen zu reduzieren.

Die dort aufgeführten Informationen stammen von den jeweiligen

Betreiber*innen oder Hersteller*innen des KI-Systems. Sie sind unter Umständen veraltet, nicht vollständig und insbesondere kritische Informationen über Systeme können in dieser Selbstauskunft fehlen, da es keine Verpflichtung gibt, negative Auswirkungen, Schwächen oder Limitationen von KI-Systemen anzugeben. Die Auswahl bildet nur einen kleinen Ausschnitt und eine Momentaufnahme der gerade existierenden Systeme ab, da KI-Systeme einerseits weiterentwickelt werden, andererseits manche Projekte eingestellt werden und möglicherweise dazu in einigen Jahren keine Informationen mehr zu finden sind. Ebenso basieren nicht alle ausgewählten Anwendungen ausschließlich auf KI. Zumeist stellt Machine Learning nur einen Teil der Technologie dar und wird mit anderen digitalen Systemen verbunden. Wenn euch dies im Einzelfall interessiert, findet ihr häufig weitere Informationen bei den jeweiligen Betreiber*innen der Systeme.

Jedes der KI-Systeme ist mit einem oder mehreren Hashtags versehen, beispielsweise **#Pflanzenkunde**, **#kostenfrei** und **#OpenSource**. So könnt ihr schnell einen Überblick bekommen, in welchem Themenfeld die Anwendung eingesetzt wird, aber auch, ob ihr diese kostenfrei selbst ausprobieren könnt und ob der Quellcode der Anwendung einsehbar ist.

Wissenschaftliche Projektleitung Dr. Theresa Züger



Hellrote Linien schneiden, hellgrüne Linien falten.



Hellrote Linien schneiden, hellgrüne Linien falten.

Impressum

Herausgebende:

Alexander von Humboldt Institut für Internet und Gesellschaft gGmbH und die gemeinnützige Organisation neuland & gestalten gGmbH

Spielentwicklung und -design:

Büro für Sinn und Unsinn GbR Illustrationen und Portrait-"Fotos" erstellt mit der KI Midjourney

Lizenz:

Creative Commons BY-NC-SA 4.0. Du darfst das Spiel für nicht-kommerzielle Zwecke kopieren und bearbeiten, solange du die ursprünglichen Urheber*innen nennst und die Lizenz beibehälst und nennst. Die PDF-Dateien sind als OER verfügbar unter: hiig.de/KI-Kompass-download www.neulandundgestalten.de/ki-kompass

Spiele bestellen:

Bildungsträger können das Spiel kostenfrei bestellen, so lange der Vorrat reicht. Schreib dazu an: info@buerofuersinnundunsinn.de





Büro für Sinn und Unsinn