

AI COMPASS: Print-and-Play Game Materials on A4 paper

Print the entire PDF on standard paper.

Glue the fronts and backs together (use a glue roller if available, or alternatively a glue stick or similar).

Cut out the cards (use a paper cutter if available, or alternatively a ruler and craft knife).

Cut out, fold, and glue the game box (if available, preferably on thicker paper, 250g).

This PDF also includes blank cards for your own ideas. For your own card designs :)



AI Joker

You can use this card for an AI you know. The AI Joker remains the same AI until the end of the game.

You want to design your own AI cards for your game? Here are the PDF templates:



CycleAI

Finds the safest and fastest bike route from A to B. Safety is assessed using collaboratively generated "hotspot maps." It was developed locally for Lisbon.

#sports
#navigation
#traffic
#free



Flora Incognita

Identifies about 16,000 species of plants from photos. Saving your own data also supports research, for example, on biodiversity.

#botany
#free



DeepL

Translates entire texts from and into different languages. Currently, 31 languages are available, including Chinese, Japanese, Indonesian, and Russian as well as nearly all official EU languages.

#translation
#free
#opensource



Mycroft

This assistance system can use voice commands, for instance, to play music and provide real-time information, create to-do lists, or control smart home systems.

#assistance
#opensource



SiMAX

Translates text-based contents of, for example, films, websites, museum tours, or citizens information, into 3D animated sign language using a virtual avatar.

#translation



hunter.io

Finds professional email addresses of people within companies.

#addresssearch
#free



Plantix

The app helps farmers to recognise (agricultural) plant diseases via a photo, gives tips for treatment, and thus increases crop yields.

#botany
#free
#opensource



Dall-E

Creates images based on text input. The generated images can be photo-realistic or imitate a certain style and can be modified, for example, by inserting an element.

#imagegeneration
#free





Automatically detects and marks which words in a text should be rewritten to be gender neutral.

#textassistance
#free
#opensource



Makes it possible to control apps with your eye movement. It allows you to play music or use devices even if, for example, you can't use your hands.

#assistance
#opensource



Helps nonnative speakers to learn to speak English without an accent by providing instant feedback on pronunciation.

#languageassistance
#free



Stockfish is a powerful chess AI algorithm that plays the game at world champion level.

#entertainment
#free
#opensource



Can predict which trees in a city will be most at risk of drought, allowing local residents to intervene early by watering them.

#natureconservation
#opensource



You can use this card for an AI you know. The AI Joker remains the same AI until the end of the game.

You want to design your own AI cards for your game? Here are the PDF templates:



Supports visually impaired people, for example, by reading texts aloud, recognising and describing people, recognising products, or translating colours and light into speech and sounds.

#translation
#assistance
#free



Uses robots and object recognition to identify and remove plastic debris from the sea floor.

#natureconservation



A fitness AI that measures heart rate and analyses and predicts training conditions and success based on past data and weather conditions.

#sports





ChatGPT

Simulates human conversations in dialogue format and provides answers to users' questions. ChatGPT can also generate programming code or texts in specific styles.

more info:



#textgeneration
#free



Project Connect

Performs analyses for UNICEF on which schools across the world do not yet have internet access. The goal is to provide access to as many schools as possible.

more info:



#education
#infrastructure
#opensource



RightHear

Provides support for people with visual impairments by helping them navigate rooms via audio descriptions.

more info:



#assistance
#free
#opensource



apic.ai

Analyses bee and bumblebee behaviour to consider the influence of environmental factors, farming practices, and substances on their health.

more info:



#opensource



rainforrest connection

AI technology that enables people to monitor and protect biodiversity and the health of the rainforest.

more info:



#natureconservation
#opensource



Readow

Makes suggestions for books a user may like based on their favourite books.

more info:



#recommendationsystem
#free



brighter Redact

Allows images and videos to be anonymised, for example, by making faces or license plates unrecognisable.

more info:



#imagegeneration



KI4LSA

AI for traffic signals: Optimises traffic flow and pedestrian safety by analysing the situation on the road with sensors and changing the traffic lights accordingly.

more info:



#traffic
#infrastructure



Character AI

Allows you to talk to AI-generated characters – for example, ones inspired by famous personalities. The dialogue agents should stimulate the imagination and could also be used for language learning.

more info:



#entertainment
#free



Bird Vision

Uses cameras on wind turbines to detect and track birds and can shut down turbines in an emergency if birds are in danger.



#speciesprotection



PoliTrack

AI-supported information platform on politicians and election-related information in Germany with visualisations and data analyses.



#politicaleducation
#free



KI-Campus

Learning platform that offers broad, low-threshold, and free educational programmes on AI and fields of application, such as medicine or educational AI. The app aims to address a broad audience by offering content that introduces audiences to AI and data.



#AIeducation
#free



Claudette

Analyses the terms of use and privacy policies of online platforms to determine whether they are compliant with EU law.



#assistance
#legalcontractanalysis
#free

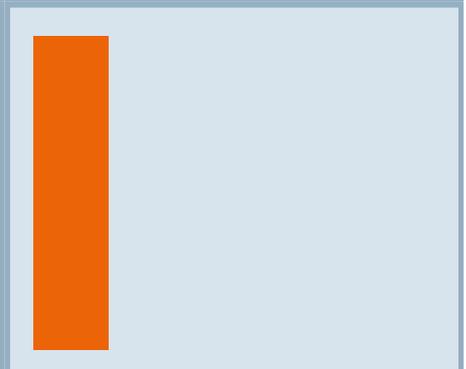
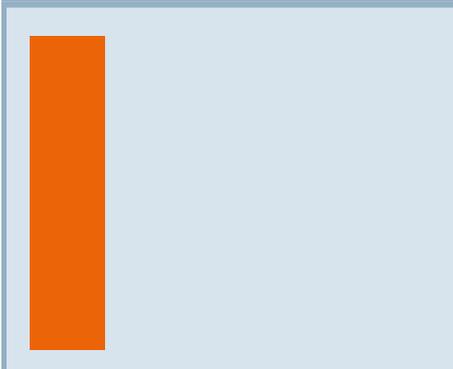


AI Joker

You can use this card for an AI you know. The AI Joker remains the same AI until the end of the game.



You want to design your own AI cards for your game? Here are the PDF templates:



Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, BatchNormalization, Activation; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, y_train, X_test, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test); model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(10)); model.compile(optimizer=Adam, loss='categorical_crossentropy', metrics=['accuracy']); model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[EarlyStopping(monitor='val_loss', patience=10)]); return model

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, BatchNormalization, Activation; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, y_train, X_test, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test); model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(10)); model.compile(optimizer=Adam, loss='categorical_crossentropy', metrics=['accuracy']); model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[EarlyStopping(monitor='val_loss', patience=10)]); return model

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, BatchNormalization, Activation; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, y_train, X_test, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test); model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(10)); model.compile(optimizer=Adam, loss='categorical_crossentropy', metrics=['accuracy']); model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[EarlyStopping(monitor='val_loss', patience=10)]); return model

```

Artificial Intelligence

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    import train_test_split
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, Batch Normalization; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); df['target'] = df['outcome']; X = df.drop('target', axis=1); y = df['target']; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping], epochs=100)
    model.evaluate(X_test_scaled, y_test)

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    import train_test_split
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, Batch Normalization; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); df['target'] = df['outcome']; X = df.drop('target', axis=1); y = df['target']; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping], epochs=100)
    model.evaluate(X_test_scaled, y_test)

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    import train_test_split
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, Batch Normalization; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); df['target'] = df['outcome']; X = df.drop('target', axis=1); y = df['target']; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping], epochs=100)
    model.evaluate(X_test_scaled, y_test)

```

Artificial Intelligence

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, BatchNormalization, Activation; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping])

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, BatchNormalization, Activation; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping])

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, BatchNormalization, Activation; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping])

```

Artificial Intelligence

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, Batch Normalization; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping])
    return model

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, Batch Normalization; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping])
    return model

```

Artificial Intelligence

```

def train_test_split(X_train, X_test, y_train, y_test):
    from sklearn.preprocessing import StandardScaler; import tensorflow.keras.models from tensorflow.keras.models import Sequential; from tensorflow.keras.layers import Dense, ReLU, Softmax, Dropout, Batch Normalization; from tensorflow.keras.optimizers import Adam; from tensorflow.keras.callbacks import EarlyStopping
    url = 'https://example.com/dataset.csv'
    df = pd.read_csv(url); target = 'outcome'; X = df.drop(target, axis=1); y = df[target]; X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler(); X_train_scaled = scaler.fit_transform(X_train); X_test_scaled = scaler.transform(X_test)
    model = Sequential(); model.add(Dense(128, input_dim=X_train_scaled.shape[1])); model.add(ReLU()); model.add(Dense(64)); model.add(ReLU()); model.add(Dense(32)); model.add(ReLU()); model.add(Dense(1)); model.add(Softmax());
    optimizer = Adam(); model.compile(optimizer=optimizer, loss='categorical_crossentropy', metrics=['accuracy']);
    early_stopping = EarlyStopping(monitor='val_loss', patience=10);
    model.fit(X_train_scaled, y_train, validation_data=(X_test_scaled, y_test), callbacks=[early_stopping])
    return model

```

Artificial Intelligence



Only 5 are needed. There are 6 included to make printing and cutting easier.

No AI Cards Backs



Only 5 are needed. There are 6 included to make printing and cutting easier.

Mission / Action Cards Fronts



I'm very concerned about our climate and our planet. So, I wonder how AI systems can support us in the climate crisis and help reduce the destruction of our planet.

Len Durovsky
Festival manager



I'm a farmer and I want to protect the bees in my region and fertilise my fields properly. How can I strike the right balance?

Best wishes,

Emilia Velasquez
Centro Agrícola 'Las Flores'

Friends are unemployed

You are too worried about your unemployed friends in this round and can't take on a mission. It's the next player's turn.



It's becoming more and more clear to me that accessibility is a very important social issue, but until now I have never really engaged with it. How could I get started?

Lil Comet
Artist and coach



The pollution of the oceans is really a huge problem. Do you have any ideas on how we can act more effectively and quickly?

Bianca Bürger
Conservationist



How can we make education accessible for everyone?

Ahmad Zachei
Global Partnership for Education

Hacker attack

Ask a player to show you their cards. You may choose one of their cards and put it in your hand.

No one draws a new AI from the deck.



It's my partner's birthday and I have no idea what to get as a present. Do you?

Best wishes,

Johanna



I have lost my job as a translator because my employer uses AI for everything now. What can I do?

Best regards,

Jayden-Blue Schmidt
Translator, activist,
and amateur gardener



I can't move my hands. I find it difficult to use most of my electronic devices. Is there a tool I can use?

Marc

Power outage

Each player who has at least one AI on the table has to surrender one. This AI goes back to the bottom of the deck.

No one draws a new AI from the deck.



The deadline for my biology paper is getting closer. Please help me to get it done in time. I feel completely unmotivated.

Best wishes,

Alba

Rapid development

You have ten seconds to look through the deck of AI cards, pick one card, and lay it out in front of you.



Interchangeability

Anyone can use an AI.

Move one place around the table clockwise. The cards laid out on the table in front of you stay where they are, but you take the cards in your hand with you.



We're getting married this year. We are both totally uncreative, but we still want cool and unique invitation cards. We've heard that there are great AIs that can help you with this?

Best wishes,

Tom & José

Bias

Biases that arise in the development of an AI can mean that it discriminates.

All players who were born in the months of November through January must give back an AI card and place it at the bottom of the deck.

No one draws a new AI from the deck. Now it's the next player's turn.



Your own mission

What problem is on your mind? Think up your own mission!

You want to have your own mission cards for your game? Here are the PDF templates:



Security risk

You've forgotten to do the risk assessment for a critical AI so you have to skip this round.

It's the next person's turn.



Mission / Action Cards Fronts



I want a green city and healthy plants and trees. What can I do to achieve this?

Best wishes,

A. Sedal
Employee at the
Environmental Agency

Sustainability

If we don't share our knowledge with each other but keep it to ourselves, we won't make much progress as a society. Open-source technologies share knowledge and can enable others to build on it. This is more sustainable for our society.

Take any AIs on the table that are not open source and place them back to the bottom of the deck of cards.

Note: free of charge ≠ open source



My brother has been under a lot of stress for a while and has been sleeping poorly. I don't know how to help him. Do you have any ideas?

Best wishes,

Amari



I am deaf and speak sign language. Today I have a date with a person who does not. How can we still communicate with each other?

Sascha

Inequality

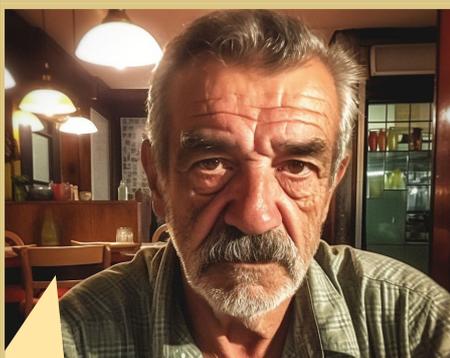
The person with the fewest AI cards laid out in front of them can draw a card from the hand of the player with the most AI cards in front of them to reduce inequality.

If there are different players to choose from, you decide.



I just got out of the hospital with my newborn, and I'm really overwhelmed. Can you support me?

Nilay Meyer



I am getting older and finding it more and more difficult to do the shopping for my restaurant. Do you have any ideas on how I can make it easier for myself?

Best wishes,

João
Restaurant owner of Joãrito

Compensation

An AI card that you have played has caused harm.

The person to your left chooses an AI card. Put it back to the bottom of the deck.

No one draws a new AI from the deck.



I have been lonely for a while. Do you have any ideas that could help me overcome my loneliness?

Best wishes,

Vian



I am just starting my own business as a designer. How do I get well paid orders?

Best wishes,
Anna Smith

Development boost

Put an AI card from your hand in front of you. Do not draw a new AI card from the deck.



I'm really interested in AI. I really want to find out more about it. Which AI could be especially fun for me?

Best wishes,

Konrad Kämmerer



My grandma is losing her sight more and more every year. I'd love to make her day-to-day life easier, but I'm rarely around. Is there anything I can do to help her?

Best wishes,
Tobia Freude

Transparency and independence

Each player may draw a new AI card. If it is open source, you can put the AI card in your hand. If it is not open source, you must put it back at the bottom of the deck.

Note: free of charge ≠ open source



Most of our teachers are completely overworked. How can we make their jobs easier?

Best wishes,

M. Mönchshammer
School Board Secretariat



I need a spring poem for my English class. And I need it in two hours!

Please help me!

Miray

New power relations

Draw one card face down from each person, shuffle the cards, and deal them again.



I want to try out my new bike. Do you guys have any tips for great places to ride to?

Best wishes,
Bilal

Mission / Action Cards Fronts



I have been trying to learn the guitar for over a year and I'm not making much progress. What can I do?

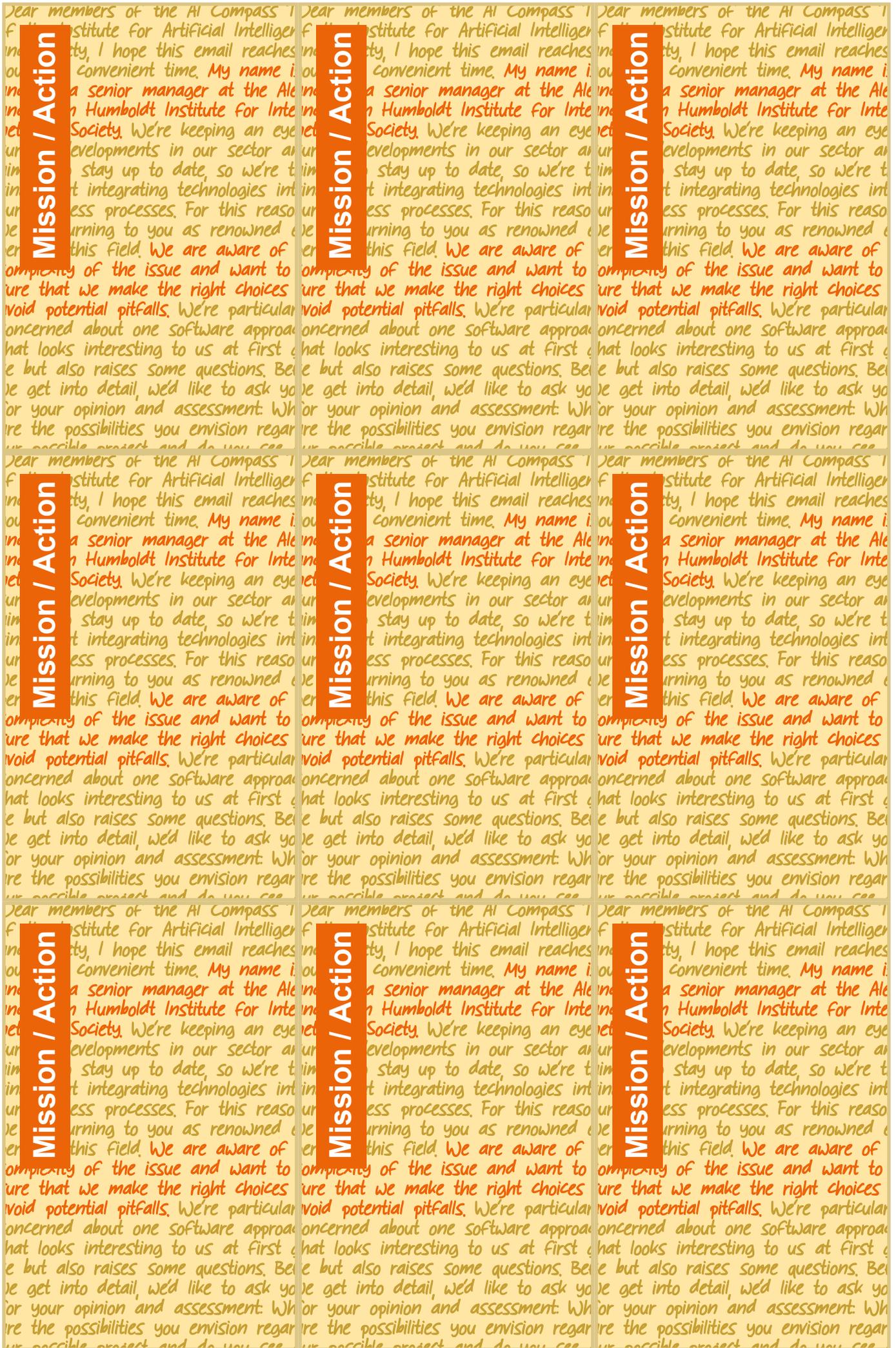
Best wishes,
Steffi



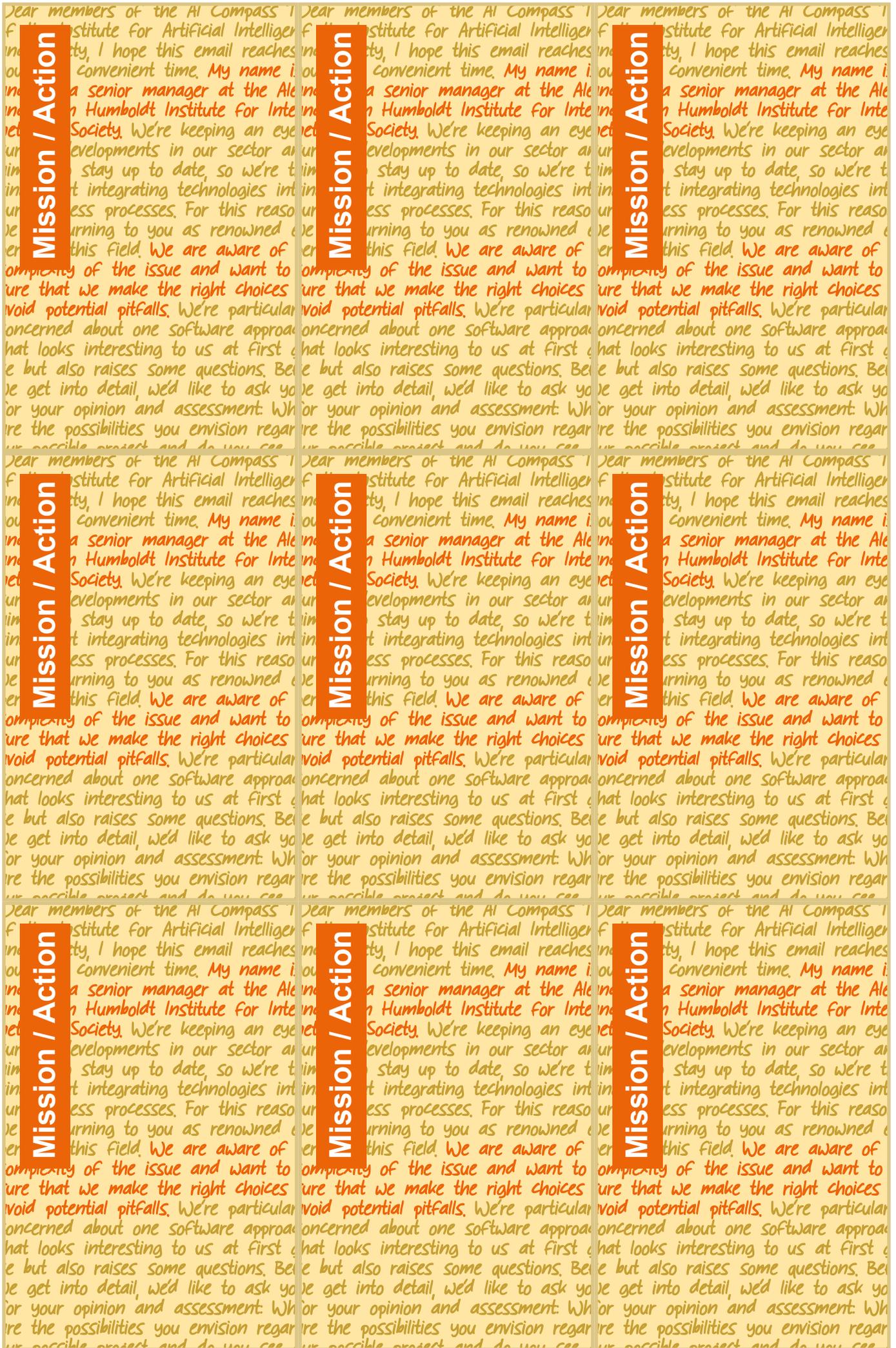
These cards are for self-labeling (optional).

These cards are for self-labeling (optional).

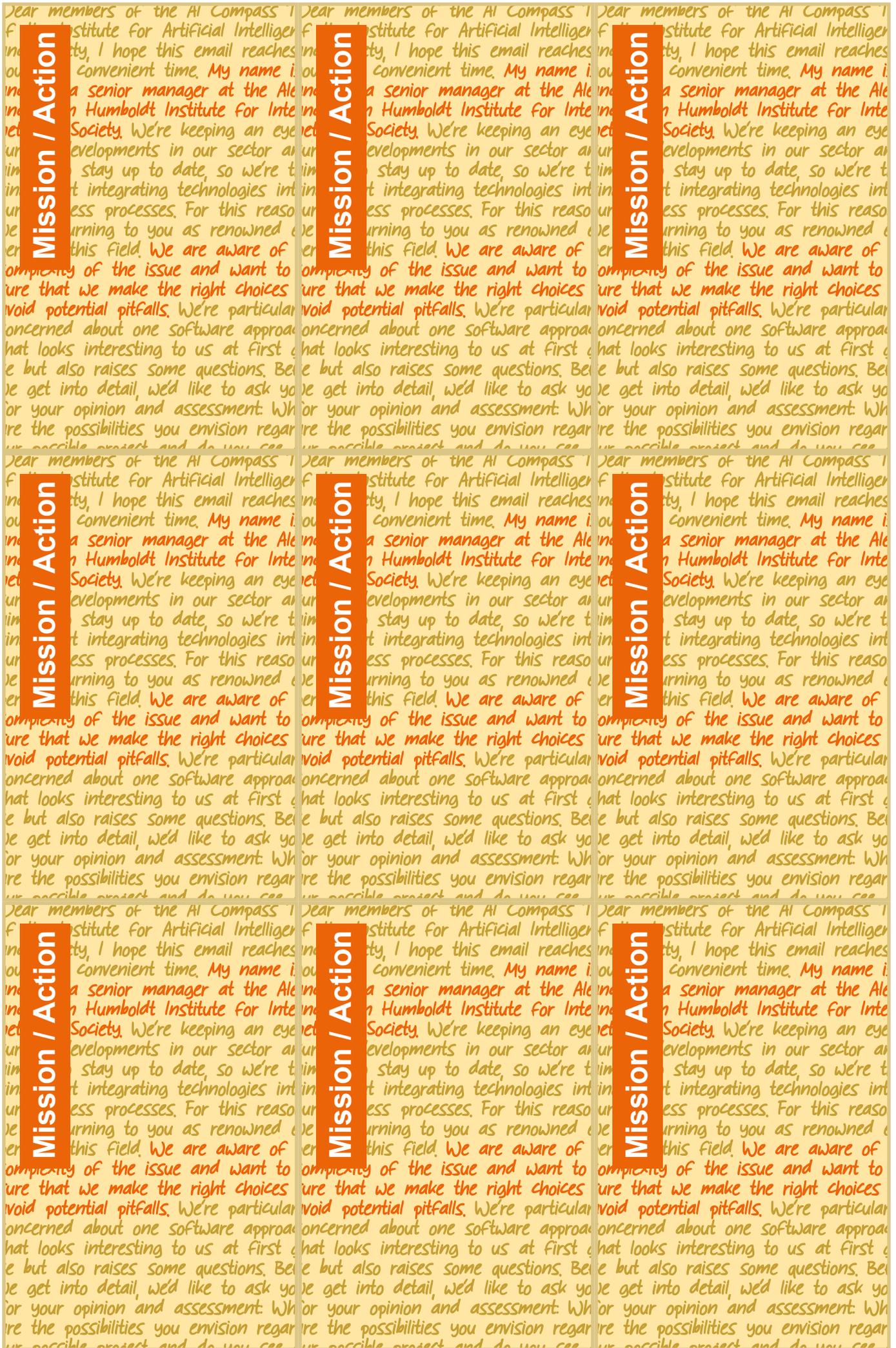
Mission / Action Cards Backs



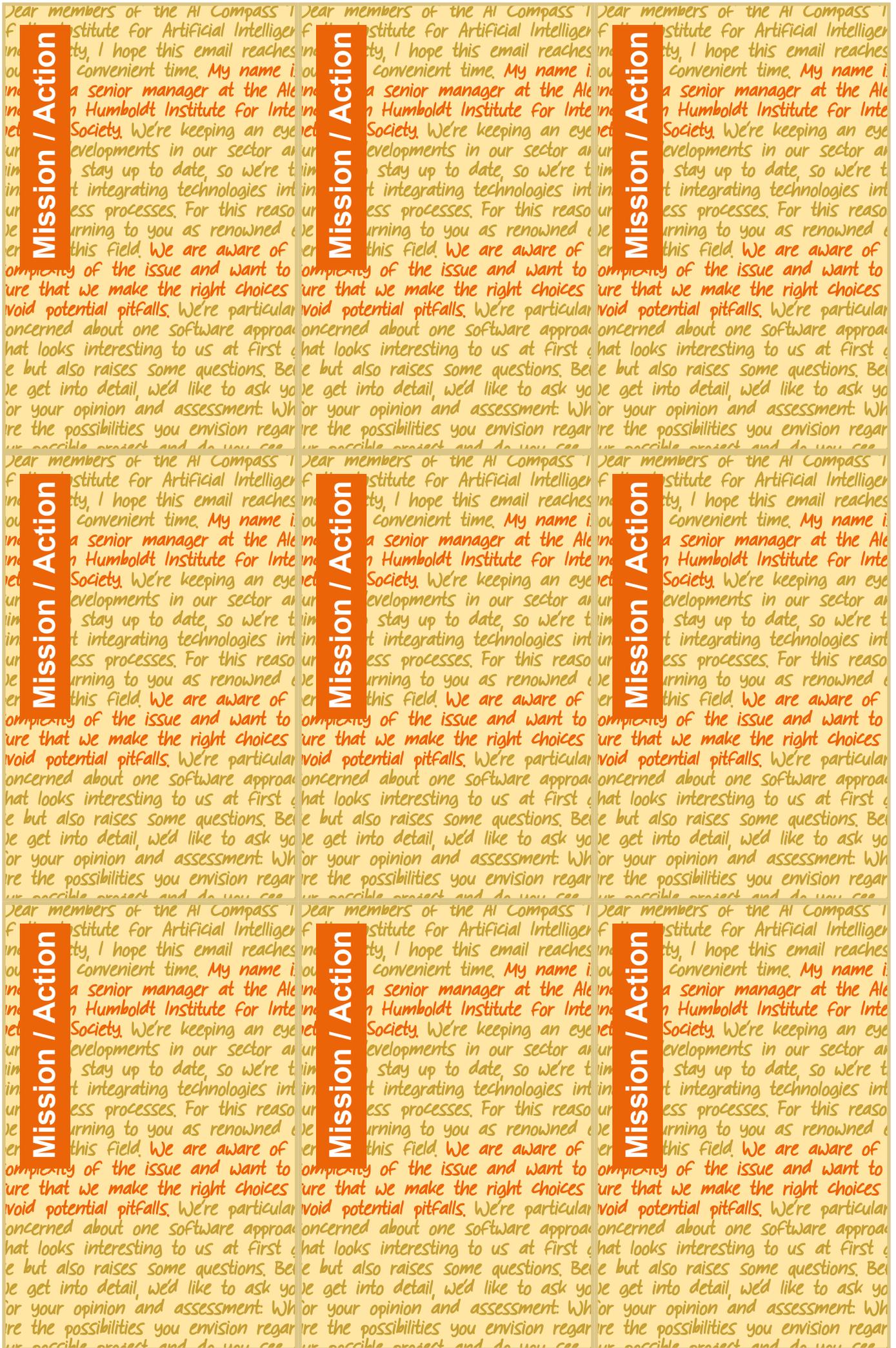
Mission / Action Cards Backs



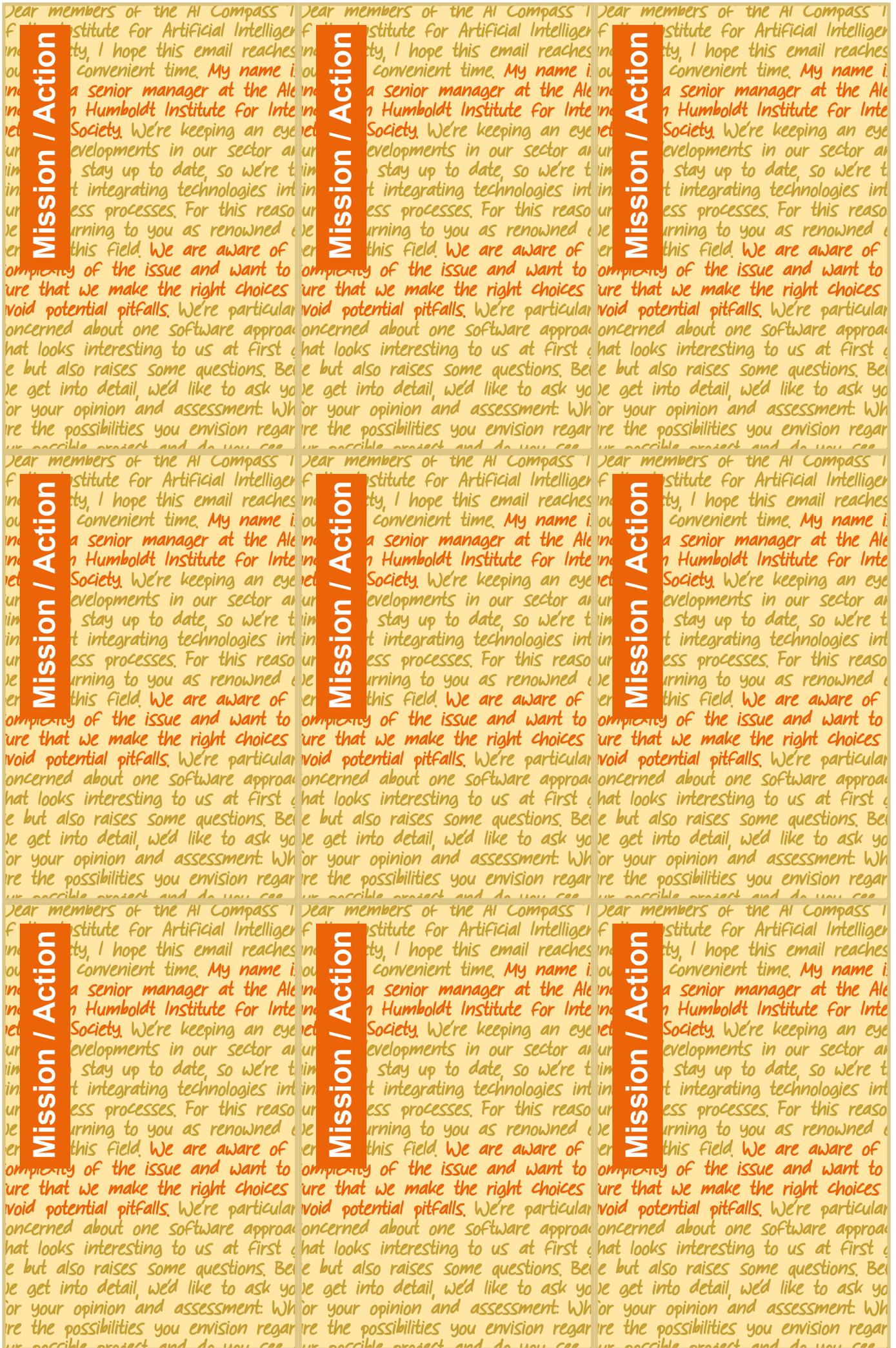
Mission / Action Cards Backs



Mission / Action Cards Backs



Mission / Action Cards Backs



About this game

The game was developed as part of the “AI Explained in Human Terms” research project, which was created through a cooperation of the **Alexander von Humboldt Institute for Internet and Society (HIIG)** with the nonprofit organization **neuland & gestalten** and implemented by the **Büro für Sinn und Unsinn**. Applying a transdisciplinary approach and working together with ordinary citizens and experts, the research project developed explanatory models to make knowledge on the topic of artificial intelligence (AI) more accessible.

More information about the project can be found at: hiig.de/en/project/ai-explained-in-human-terms

Contents

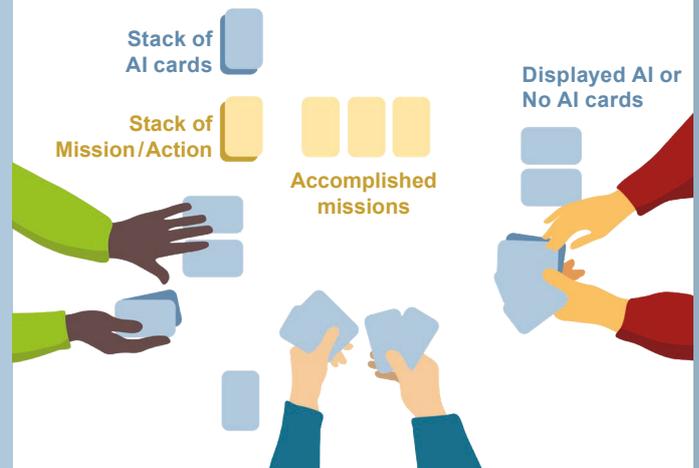
32 AI cards

5 No AI cards

37 Mission/Action cards

Game instructions

What it's about: Artificial intelligence is the talk of the town and the hype about its capabilities is huge. In this card game, you are the Artificial Intelligence Competence Team and get requests from citizens to solve problems in the form of mission cards. Together with the other members of the AI Competence Team, you will have to decide on what problems AI systems can help resolve and when you should try to find a solution without AI.



Goal of the game: Your task is to complete eight missions together. Whoever proposes the most creative and convincing solutions and has the most (AI and non-AI) cards at the end will be awarded employee of the day and win the game.

Preparation: Each player gets four **AI cards** and one **No AI card**. The remaining AI cards are placed face down in a stack in the middle. Next to it, a stack of Mission and Action cards is placed.

Rules of the game: One person begins. Turn over the top **Mission/Action card** and read the card out aloud. This card could be:

1. A Mission card: Your fellow players should each propose a solution to this problem. They can do this by placing either an **AI card** or a **No AI card** face up on the table in front of them and explaining their proposal. The team now discusses and thinks about which solution is the best one. You then decide who to award the round to.* That person may keep the card in front of them permanently from now on and should draw a new AI from the stack into their hand. All other players pick up their rejected cards and hold them in their hands. When the mission has been accomplished, the card remains face up in the middle. From the next round on, each player is not just limited to using

the cards in their hand but can also use the cards laid out in front of them. Note: If a laid-out AI card is #opensource or #free, it can be used by everyone, not just the person who laid it out!

* If you are torn between several solutions, you can, in exceptional cases, award the round to more than one player. They may then leave all of their proposed cards on the table permanently and draw one AI card each from the stack into their hands. If you don't find any of the solutions useful at all, then the mission is “not accomplished” and the card goes back to the bottom of the stack.

2. An Action card: Perform the action and then put the card back at the bottom of the stack. The next person in clockwise order now takes their turn and turns over the next card from the Mission and Action card stack.

End of the game: Keep playing the game until you have completed eight missions. The person with the most cards (AI and No AI) in front of them wins and becomes employee of the day.

What is AI?

Artificial intelligence is a complex catch-all term that is difficult to define. On the one hand, AI refers to a subfield of computer science that seeks to technically emulate human intelligence. On the other hand, how we understand human intelligence is also disputed. In today's usage, the term artificial intelligence mostly refers to various technical applications that collectively automate processes that mimic human capabilities, e.g., human vision (**computer vision**) or human language translation (**natural language processing**).

One method currently in widespread use is called **machine learning**. **Deep learning**, a subfield of machine learning that uses neural networks, is now the basis for many AI applications. Even though the term "neuron" is inspired by the neurons of the human brain, there is a difference between neurons—which are parts of the human nervous system—and technical neural networks. Neural networks are basically very large statistical systems. Data are processed mathematically, one after the other, by different nodes (neurons), which are interconnected like a network. The more complicated the task for the neural network, the more necessary it is to use more layers of neurons and a more complex network

structure. Training data is used, for example, to train neural networks to recognise an object or pattern, such as a face, or predict the most likely next word in a sentence. Although we often speak of machine learning, the term "learning" is misleading, because there is no conscious acquisition of knowledge or thinking on the part of the machine as there is in humans; the systems are just trained to recognise certain patterns. Most of the time, the results they generate take the form of a probability, so, for example, they say that an array of pixels on an image has a 90% probability of representing a human face. The exact patterns by which the neural network "recognises" the characteristics of a face are not always easy to understand. However, the field of explainability of AI within computer science uses various mathematical methods to try to understand as precisely as possible why a neural network arrives at a certain result.

The explainability of AI systems is particularly important when they are used in contexts where decisions have consequences for humans. When AI systems are used, human actors are always involved as well: They are involved in their development, in the selection of training data, in the way they are used, and, of course, in the evaluation and use of AI-generated probabilities. AI use is therefore always also a politi-

cal issue. If you are interested in the use of and politics around AI, please feel free to visit **HIIG** and the **Digitaler Salon**, where we publicly discuss issues around digital politics and AI.

About the AI systems in this game

For the AI Compass, we have selected a number of currently existing AI applications. You can find further information for each of them via the QR code on each AI card. The apps we selected do not represent the mainstream of today's AI systems: We deliberately chose them to show that AI is used in very different areas of society. Many examples illustrate that AI systems are also used in the public interest, e.g., in nature conservation or to reduce a wide variety of everyday barriers for people. The supporting information come from the operators or manufacturers of the respective AI system. It may be outdated or incomplete, and it may not include critical information about systems, as AI system developers are not required to disclose negative impacts, weaknesses, or limitations of their systems. The AI apps we selected are just a small sample and a snapshot of the currently existing systems, since AI systems continue to be developed and some projects are discontinued so there may not be any information available about them in a few years.

Additionally, not all of the selected applications are based exclusively on AI. Oftentimes, machine learning is only one part of the technology and is combined with other digital systems. If you are interested in a particular case, you can usually get more information from the operators of the systems in question.

Each of the AI systems has one or more hashtags, such as **#botany**, **#free**, and **#opensource**. This means you can quickly relate the subject area the application is used in and also find out whether you can try it out yourself for free as well as view the app's source code.

Scientific project management Dr. Theresa Züger



Cut along the red lines, fold along the green lines.



Cut along the red lines, fold along the green lines.

Imprint

Publishers:

Alexander von Humboldt Institute for Internet and Society gGmbH
and the nonprofit organization neuland & gestalten gGmbH

Game development and design:

Büro für Sinn und Unsinn GbR
Illustrations and portrait "photos" created with the AI app Midjourney

License:

Creative Commons BY-NC-SA 4.0. You may copy and edit the game for noncommercial purposes as long as you credit the original creators and retain and cite the license. The PDF files are available as open educational resources (OER) at:

hiig.de/en/ai-compass

neulandundgestalten.de/ki-kompass



Büro für Sinn und Unsinn