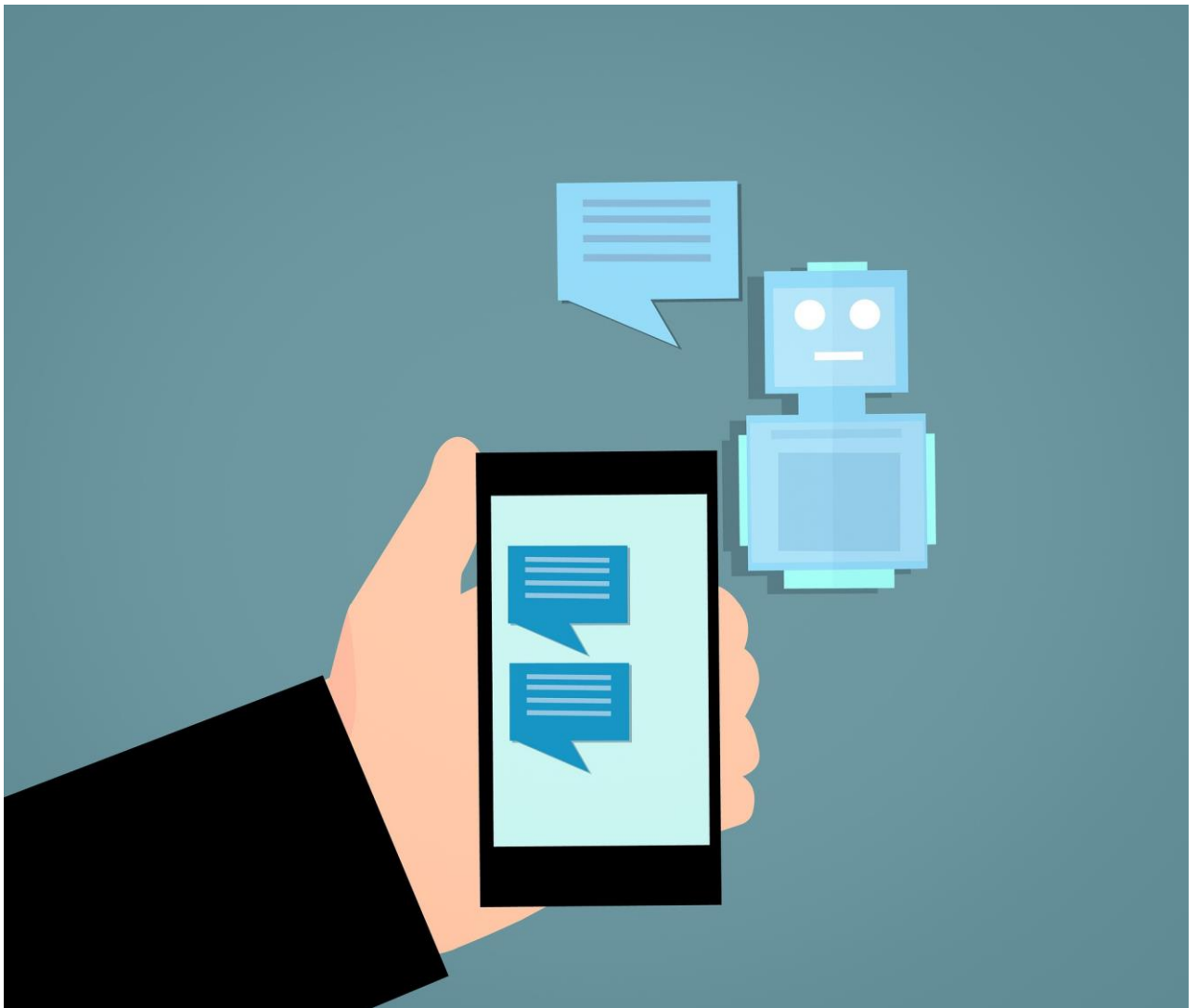


Chatbot programmieren

Ferienkurs an der FH-SWF



Chatbot programmieren

Was ist ein Chatbot?

Das Wort Chatbot setzt sich aus „chat“ (Englisch für plaudern) und „bot“ (Abkürzung von „robot“) zusammen. Diese sind in erster Linie dazu da uns auf einfache Art und Weise in einem Gespräch zu helfen, oder uns über bestimmte Themen aufzuklären. Sie sind im Grunde ein Computerprogramm, welches die Interaktion zwischen Mensch und Maschine ermöglicht. Oft werden Bots im Servicebereich genutzt, um Kunden einfache Fragen zu beantworten, für die normalerweise kein Kontakt zu einem Mitarbeiter benötigt wird. Mittlerweile werden die Bots jedoch auch in anderen Bereichen, wie dem Erstkontakt mit Studieninteressierten an Hochschulen, genutzt. Ein Beispiel dafür ist Marvin, der im Rahmen eine Bachelorarbeit von einer Studentin an der FH-SWF entwickelt wurde und bald für die Studienberatung eingesetzt werden soll.

Chat mit Marvin

Hallo, ich bin Marvin. Wie kann ich Dir helfen?

less than a minute

Halo Marvin, ich möchte gerne Informatik studieren. Geht das an der FH-SWF?

less than a minute

Informatik (B.Sc.)\nSchaff die Basis, um später als IT-Fachkraft die Welt im Zukunftsprojekt Industrie 4.0 und die Technologien von morgen mitzugestalten. Den Anfang machst du am besten hier: https://www4.fh-swf.de/de/home/studieninteressierte/studienangebote/stg_is/informatik/index.php

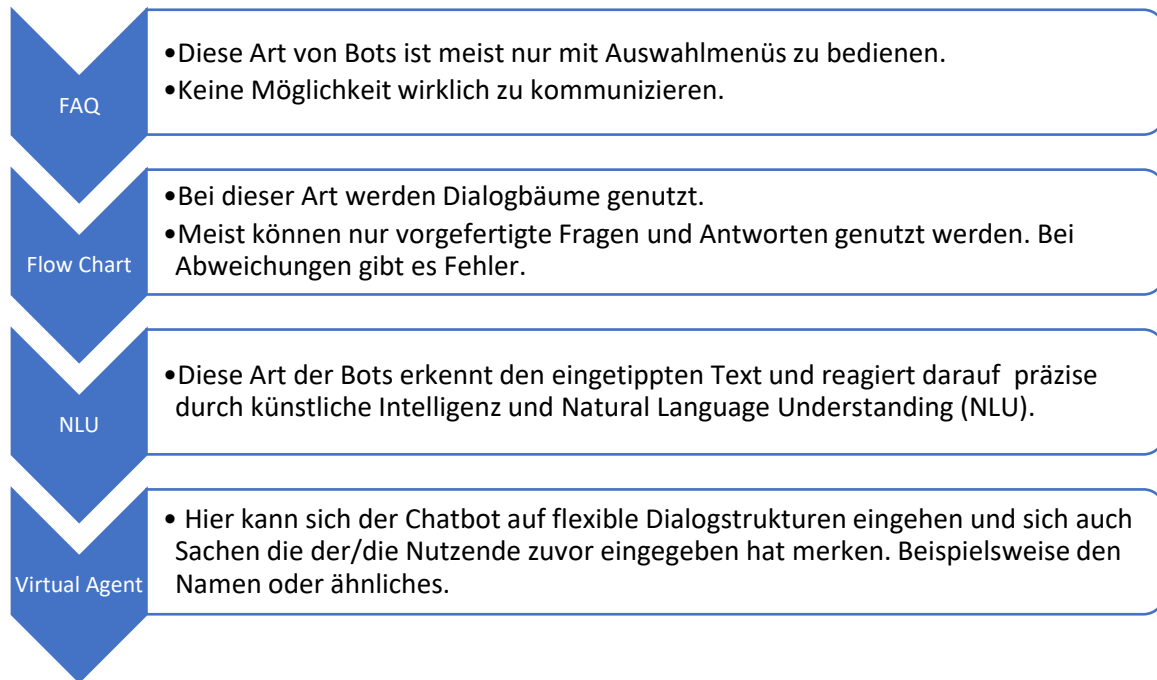
less than a minute

SUBMIT

Um sich mit dem System unterhalten zu können, wird überwiegend eine künstliche Intelligenz verwendet. So können Fragen gestellt werden, auf welche die Anwendung in natürlicher Sprache antworten kann. Meist gibt es neben der Texteingabe auch noch die Möglichkeit, dem Chatbot per Audioeingabe Fragen zu stellen. Ein berühmtes Beispiel für so eine künstliche Intelligenz ist Amazons Alexa.

Gibt es Unterschiede bei Chatbots?

Chatbot ist nicht gleich Chatbot. Es gibt mehrere Varianten von Bots, die jedoch sehr unterschiedlich in ihrer „Intelligenz“ sein können.



Was wird hier genutzt?

Der Beispielbot basiert auf Rasa und kann euch viele eurer Fragen zum Franchise „Pokémon“ aus dem Hause „Nintendo“ beantworten. Er ist im Vorfeld schon mit ein paar Beispieldaten zu dem Thema „gefüttert“ worden.

Auf dieser Basis könnt ihr durch die einzelnen Aufgaben echte Chatbotprofis werden!

Ihr erfahrt:

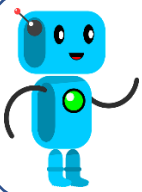
- Wie ein Chatbot funktioniert.
- Wie intelligent ein weitestgehend untrainierter Bot wirklich ist.
- Aus welchen Komponenten ein Chatbot besteht.
- Was einen Chatbot von Rasa ausmacht.
- Mit was man ihn „füttern“ kann.

Was ist ein Rasabot?

Rasa ist ein Open-Source-Tools für Entwickler zur Erstellung von Text- und Sprachassistenten auf Basis von maschinellem Lernen. Der Bot lässt sich in der Kategorie „NLU“ einordnen.

Welche Komponenten dieses System genau ausmachen, lernen wir im Laufe dieser Lehrinheit.

Wie intelligent ist der Chatbot?



Hier könnt ihr nun testen, wie intelligent der Chatbot wirklich ist. Versucht die Reaktionen zu verstehen.

1. Redet mit dem Chatbot über Pokemon. Fragt ihn nach den Regionen, wie viele Pokemon es gibt und fragt ihn nach eurem Starterpokemon! Er weist euch gerne eins zu. Schreibt eure Beobachtungen auf. Versucht noch weitere Themen zu finden. Habt ihr ihn etwas zu Pokemon gefragt und er hat es nicht verstanden?

2. Was passiert, wenn ihr Eingaben wiederholt? Antwortet er gleich?

3. Ändert nun die Formulierung von Sätzen, die ihr im Vorfeld schon eingegeben habt. Ändert sich die Antwort? Versteht er noch, worum es geht?

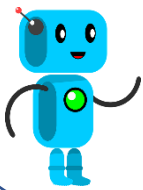
4. Versucht herauszufinden wie viele Pokémon der Chatbot kennt. Kennt er alle 906 Pokémon? Welche kennt er?

5. Findet das Lieblingspokémon vom Chatbot heraus!

6. Müsst ihr in ganzen Sätzen antworten, oder reagiert er auch bei Stichwörtern?

7. Was passiert, wenn ihr mit ihm über das Wetter reden wollt oder ihr nach einem Witz fragt?

8. Tauscht euch nun in Gruppen oder im Plenum über eure Ergebnisse aus. Habt ihr alle die gleichen Erfahrungen machen können? Gibt es gravierende Unterschiede in den Ergebnissen? Wenn ja, wo genau?



Ihr könnt euren Chatbot personalisieren! Dazu hier nochmal ein paar Grundlagen zu dem Rasa Chatbot.

Der Bot besteht aus mehreren Komponenten. Die wichtigsten Komponenten für heute sind die folgenden:

1. nlu.yml

In der Datei nlu.yml habt ihr die sogenannten Intents (zu Deutsch, absichten). Hier sind alle möglichen Fragen, die ein User stellen könnte, hinterlegt.

Diese liegen in folgender Form vor:

```
- intent: ash
  examples: |
    - Wer ist Ash?
    - Erzähl mir was über Ash?
    - Wer ist der Protagonist?
    - Wer ist Ash Ketchum?
    - Woher kommt Ash?
    - Wer ist Ash Mutter?
    - Wie heißt der Protagonist?
    - Wer ist der Hauptcharakter?
    - Wie heißt Ash mit Nachnamen?
    - Was ist Ashs Nachname?
    - Wer ist Ashs Mutter?
    - Wie heißt Ashs Heimatstadt?
    - Wer reist in der Pokemonserie herum?
    - Wer ist Pikachus Trainer?
```

Wichtig ist, dass ihr genau die oben abgebildete Syntax einhaltet! Hinter dem Intent vergebt ihr einen passenden Namen. In diesem Beispiel heißt dieser „ash“, weil es Fragestellungen zu dem Pokemon Protagonisten enthält. Danach gebt ihr mit dem gleichen Abstand wie oben, „examples: |“ ein. Das Pipezeichen (|) muss eingegeben werden, sonst funktioniert es nicht!

Danach gebt ihr immer mit einem Minuszeichen voran, eure möglichen Fragestellungen ein. Hier müsst ihr genau überlegen und euch in den/die Nutzende hineindenken. Auf wie viele verschiedene Arten könnte die Frage gestellt werden?

2. domain.yml

In dieser Datei müsst ihr zunächst zu anfang nochmal eure gesamten Intents aus der nlu.yml in folgender Form angeben:

```
intents:
  - greet
  - goodbye
  - affirm
  - deny
  - didnt_understand
  - mood_great
  - mood_unhappy
  - bot_challenge
  - demand_joke
  - what_is_pokemon
  - how_many
  - confirm_exist
  - ask_berrries
  - ash
  - pikachu
  - history
  - types
  - forms
  - regions
  - favorite_pokemon
  - poke_partner
```

Solltet ihr einen neuen Intent in die nlu-Datei einfügen, müsst ihr diese auch unbedingt hier hinterlegen! Die Domain-Datei ist dazu da, dass zwischen den Eingaben des Nutzens und den Ausgaben des Bots eine Verbindung hergestellt werden kann.

Hier werden nämlich auch die einzelnen Antworten des Bots hinterlegt. Die Antworten starten meist mit einem „utter“. Bitte bedenkt, dass ihr keine Leerzeichen in dem Namen einfügen könnt. Nehmt stattdessen Unterstriche also „_“. Zu Beginn startet ihr mit „responses:“:

```
responses:
  utter_greet:
    - text: "Hey, wie gehts dir? Möchtest du etwas zu Pokemon wissen?"
    - text: "Hallo, wie gehts?"
    - text: "Guten Tag, wie geht es dir?"
    - text: "Hallöchen, wie gehts?"
    - text: "Hallo, kann ich dir eine Frage zu Pokemon beantworten?"

  utter_cheer_up:
    - text: "Hier ist etwas, was dich aufheitern sollte:"
      image: "https://i.pinimg.com/originals/65/15/ff/6515fff25bdd8f04cbf3e5dbd4a4ed00.png"
```

Zu einem Thema können auch mehrere Antwortmöglichkeiten hinterlegt werden. Der Bot sucht sich dann zufällig eine davon aus. Die Antworten müssen stets in Einführungsstriche gesetzt werden. Jedes weitere Thema auf, welches der Chatbot antworten kann, wird direkt mit der jeweiligen Überschrift daruntergesetzt.

Die Antwort zu der Frage wer Ash ist, wird hier in folgender Form hinterlegt:

```
utter_faq_pikachu:
- text: "Ash erhält Pikachu in der ersten Episode des Animes von Professor Eich,
  obwohl er die Verteilung der Starter-Pokémon eigentlich verpasst hat.
  Pikachu hasst es in seinem Pokéball zu bleiben und kann Ash zu Beginn ihrer gemeinsamen Reise
  nicht leiden.
  Trotzdem werden sie im Laufe ihrer Reise beste Freunde.
  Pikachu lehnt es ab, sich weiterzuentwickeln, denn seiner Meinung nach würde es dann seine
  Kraft alleine dem Donnerstein verdanken.
  Außerdem wäre dies die Bestätigung, dass Raichu generell Pikachu gegenüber überlegen seien,
  was nicht seiner Auffassung entspricht. (Pokewiki)"
```

Sobald ihr alle Antwortmöglichkeiten zu euren Fragethemen eingefügt habt, könnt ihr euch die nächste Art von Datei anschauen.

3. Stories.yml

In dieser Datei hinterlegt ihr mögliche Gesprächsszenarien. Stellt euch vor, ihr würdet mit einer anderen Person sprechen. Überlegt euch wie ihr jeweils als Benutzende*r Fragen stellen würdet und wie der Bot antworten würde.

In dem folgenden Beispiel ist die Story zu dem jeweiligen Pokepartner noch einmal genauer erklärt:

<pre>- story: poke partner happy steps: - intent: greet - action: utter_greet - intent: poke_partner - action: utter_poke_partner - action: utter_poke_choose - action: utter_satisfied - intent: affirm - action: utter_happy - action: utter_good_ask_rephrase</pre>	<pre>Your input -> Hallo Hey, wie gehts dir? Möchtest du etwas zu Pokemon wissen? Your input -> Ja, was wäre mein Starterpokemon? Dein Pokepartner wäre: Glumanda Bist du zufrieden, mit meiner Wahl? Your input -> Ja, Glumanda ist super! Super! Falls du eine Frage hast, stell sie ruhig.</pre>
--	--

Diese Story würde genau dem vorgegebenem Storypfad entsprechen.

4. Rules.yml

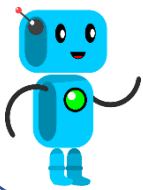
Es gibt auch noch Möglichkeiten Regeln vorzugeben. Dies wäre zum Beispiel der Fall, wenn man auf eine Frage immer eine feste Antwort erhalten möchte. Man weiß in diesen Fällen, dass es zu keinem anderen Gesprächsfluss kommen kann. So wird der Bot einen stets verabschieden, wenn ein Abschiedsgruß vom User gesendet wird.

```
version: "3.1"

rules:

- rule: Say goodbye anytime the user says goodbye
  steps:
  - intent: goodbye
  - action: utter_goodbye
```

Wahrscheinlich habt ihr schon ein bestimmtes Muster bei der Syntax der Dateien erkannt. Jetzt wo wir die wichtigsten Komponenten einmal gesehen haben und wissen wie diese aussehen, können wir auch schon selbst Änderungen an dem Chatbot durchführen!



Hier habt ihr auch noch eine Übersicht zu euren Zugangsdaten und den wichtigen Vorgängen auf der virtuellen Maschine

1. Benutzerdaten

Eure Zugangsdaten für die virtuelle Maschine lauten:

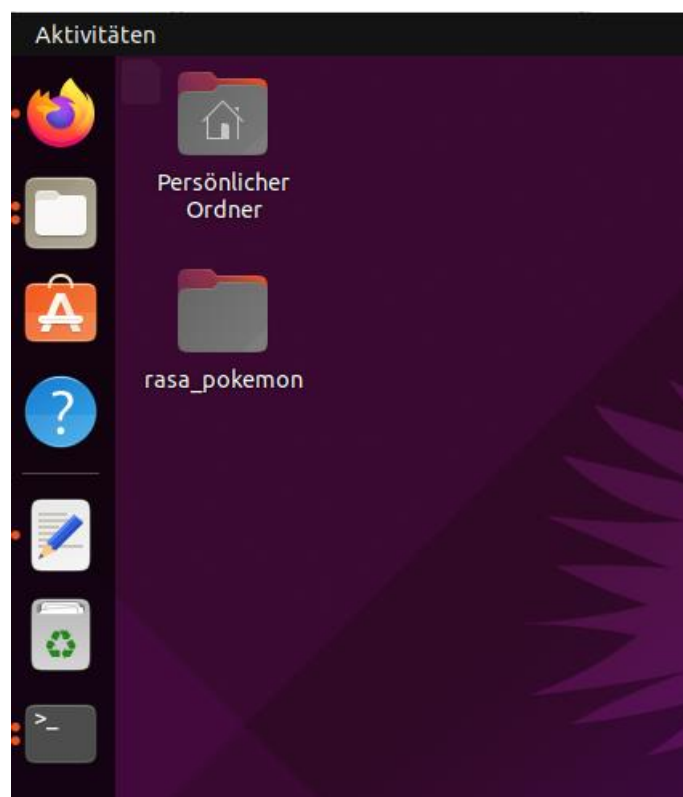
Benutzer: student

Passwort: student

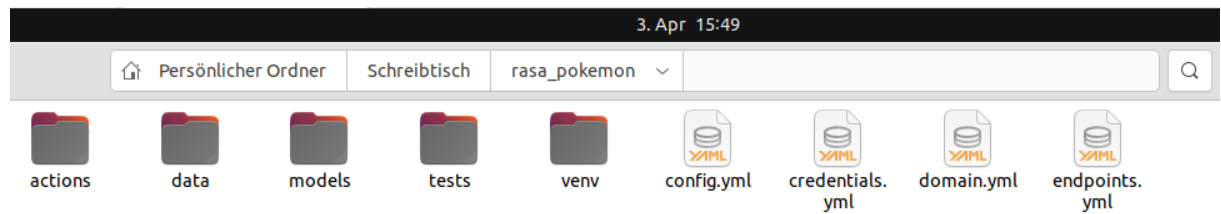
In der virtuellen Maschine läuft ein Betriebssystem namens Linux Ubuntu. Falls ihr Linux noch nicht kennt, könnte es sein, dass ihr einen Moment braucht, um euch zurecht zu finden.

2. Projektordner

Auf dem Desktop der Maschine sollte euch der Ordner „rasa_pokemon“ angezeigt werden.

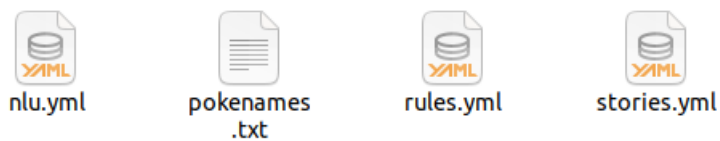


Mit Doppelklick auf den Ordner, öffnet sich eure Projektumgebung im Dateieexplorer.

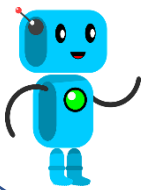


Die domain.yml findet ihr auf dieser Ebene.

Die nlu.yml sowie die stories.yml und die rules.yml findet ihr im Ordner „data“.



Um die Dateien zu ändern, klickt ihr einfach mit einem Doppelklick auf die jeweilige Datei. Diese öffnet sich dann in einem Textbearbeitungsprogramm und kann von euch geändert und angepasst werden.



Nun personalisieren wir endlich unseren Pokémon Chatbot.
Habt die vorherigen Seiten am besten dafür stets geöffnet.

1. Das Thema „Formen“ hinzufügen

Wie ihr vielleicht wisst, gibt es mehrere Formen von Pokemon.

Das kann unterschiedliche Gründe haben. Beispielsweise die Dynamaximierung, verschiedene Geschlechter, oder aber auch einfach je nach Gebiet, wo man die Pokemon findet. Falls ihr das Phänomen noch nicht kennt, habt ihr hier einmal das Beispiel von einem Raichu aus Kanto (links) und einem aus Alola (rechts).



Es handelt sich um das gleiche Pokemon aus verschiedenen Regionen. Genau dieses Thema möchten wir nun unserem Chatbot beibringen. Dazu fügt ihr in der Datei nlu.yml folgenden Intent hinzu. Ihr könnt statt „forms“ auch „formen“ oder ähnliches schreiben. Bedenkt aber dann, dass ihr auch in allen anderen Dateien die neue Schreibweise nutzt. Sonst funktioniert das Training des Chatbots nicht!

- **intent: forms**
- examples: |**
 - Gibt es verschiedene Pokemonformen?
 - Hat ein Pokemon mehrere Formen?
 - Gibt es regionale Formen?
 - Können Pokemon mehrere Formen haben?
 - Gibt es Pokemon mit mehreren Formen?

Ihr könnt gerne noch ein paar andere Formulierungen nach dem Muster einbringen. Lasst eurer Kreativität freien Lauf! Denkt aber daran jede Datei nach der erfolgreichen Änderung zu speichern!

Danach muss der neue Intent auch zu den Intents in die domain.yml. Diesen fügen wir am besten am Schluss hinzu, um einen Überblick über die Neuerungen zu haben.

```
intents:
  - greet
  - goodbye
  - affirm
  - deny
  - didnt_understand
  - mood_great
  - mood_unhappy
  - bot_challenge
  - demand_joke
  - what_is_pokemon
  - how_many
  - confirm_exist
  - ask_berrries
  - ash
  - pikachu
  - history
  - types
  - regions
  - favorite_pokemon
  - poke_partner
  - forms ←
```

Nachdem dieser erfolgreich eingefügt wurde, müssen wir uns noch um die Antwort des Bots auf das Thema kümmern. Hier könnt ihr gerne googlen, euren eigenen Wortlaut einfügen, oder folgende Formulierung nutzen:

```
utter_forms:
  - text: "Es gibt mehrere Arten zu unterschiedlichen Formen. Es gibt zum Beispiel die Gigadynamaximierung, Megaentwicklung, Regionalformen, Formwandler und Formen durch den Fangort oder durch verschiedene Items die man vergeben hat."
```

Nachdem die Datei gespeichert wurde, öffnen wir nun die stories.yml. In dieser legen wir jetzt einen Gesprächsverlauf fest. Denk daran auch hier eure Formulierungen genau zu übernehmen! Sonst kann der Chatbot die Story nicht erkennen.

- **story: forms**
 - steps:**
 - **intent: greet**
 - **action: utter_greet**
 - **intent: forms**
 - **action: utter_forms**
 - **action: utter_did_that_help**

Die Stories fangen wir stets mit einer Begrüßung des Nutzenden an. Deshalb hier „intent: greet“, darauf antwortet dann der Bot. Nach der Begrüßung fragt der User nach den Pokemonformen. Der Chatbot antwortet darauf mit unserer festgelegten Antwort. Danach wird noch gefragt, ob das hilfreich war. Die Story kann hier beendet werden. Der Chatbot lernt auch noch durch Training mögliche Verläufe.

Herzlichen Glückwunsch! Ihr habt nun euer erstes eigenes Thema in den bestehenden Chatbot integriert! Nach einem erneuten Training des Bots, könnt ihr eure aktuelle Version testen. Probiert es doch direkt mal aus!

```
Your input -> Hallo
Hey, wie gehts dir? Möchtest du etwas zu Pokemon wissen?
Your input -> Ja, gibt es verschiedene Formen von Pokemon?
Es gibt mehrere Arten zu unterschiedlichen Formen. Es gibt zum Beispiel
die Gigadynamisierung, Megaentwicklung, Regionalformen, Formwandler un
d Formen durch den Fangort oder durch verschiedene Items die man vergebe
n hat.
Konnte ich helfen?
```

2. Eine zweite Antwort hinzufügen

Wir wissen jetzt, wie wir dem Chatbot neue Themen beibringen können. Zunächst fügen wir jedoch noch eine zweite Antwort zu unserer vorherigen Frage hinzu.

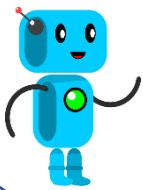
Dazu öffnen wir noch einmal die Datei domain.yml und scrollen zu „utter_forms“.

Hier ergänzen wir eine weitere Botantwort.

```
utter_forms:
- text: "Es gibt mehrere Arten zu unterschiedlichen Formen. Es gibt
zum Beispiel die Gigadynamisierung, Megaentwicklung, Regionalformen,
Formwandler und Formen durch den Fangort oder durch verschiedene Items
die man vergeben hat."
- text: "Es gibt bei Pokemon viele Arten von Formen. Beispielsweise
können gleiche Pokemonarten aus verschiedenen Regionen anders aussehen.
In Alola sind 18 Pokémon mit Regionalformen bekannt, die allesamt
ursprünglich aus Kanto stammen. Es gibt aber auch noch mehrere Arten von
Entwicklungen oder ähnlichem was das Aussehen beeinflussen können."
```

1. Fragt mehrmals nach den verschiedenen Pokémonformen. Erscheint immer die selbe Antwort? Wie verhält sich der Bot?

2. In der Story haben wir nach dem „Konnte ich helfen“ nicht weiter gemacht. Kann der Bot trotzdem noch andere Fragen beantworten?



Jetzt habt ihr die Chance eure eigenen Themen in den Chatbot zu integrieren. Was wird es wohl werden?

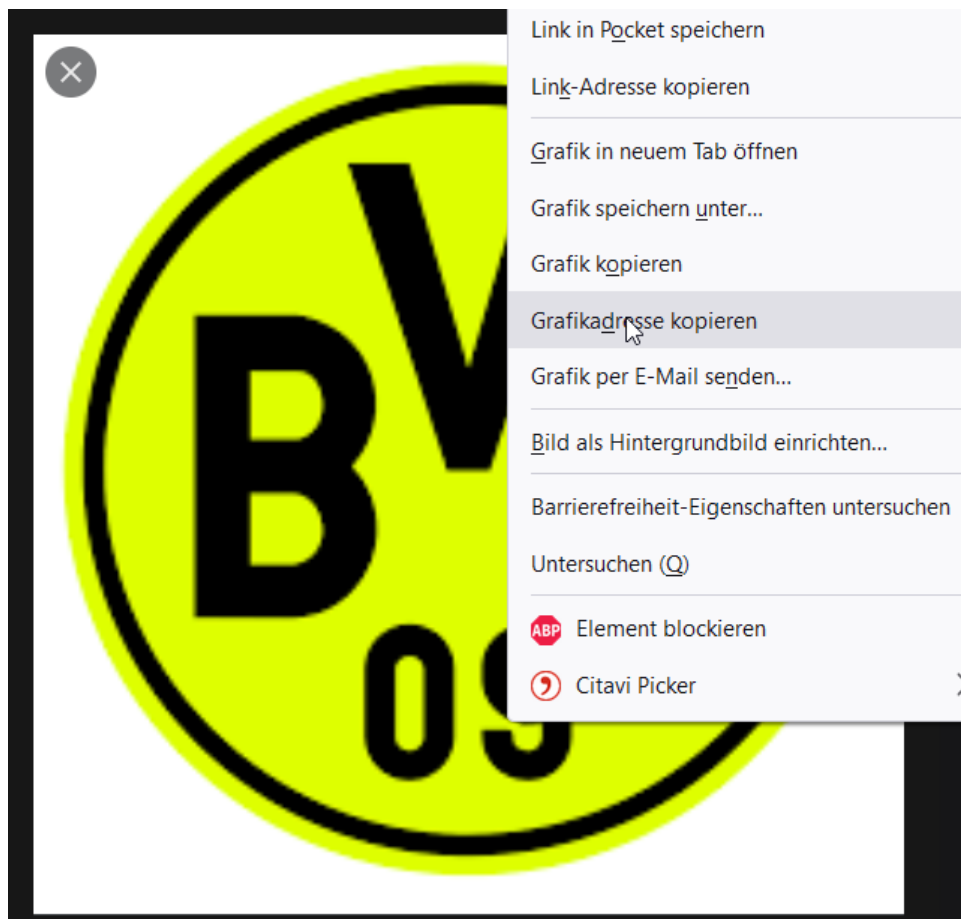
Tipps:

Nachdem wir jetzt einmal alle relevanten Funktionen durchgearbeitet haben, ist es nun an der Zeit ein paar eurer eigenen Themen zu implementieren.

Ihr müsst keinen Inhalt zu Pokemon wählen. Wie wäre es vielleicht mit Fragen zu eurem Wohnort, der Fachhochschule, eurem Lieblingsschulfach, oder einem Hobby von euch? Denkt dran jeden Schritt wie zuvor zu einem Thema bzw. einer Frage abzuarbeiten.

Zusatzaufgaben:

- 1) Fügt der Antwort eures Bots einen Link zu einem Bild hinzu. Sucht dazu in einer Suchmaschine eurer Wahl nach einer Abbildung. Im Browser ist es möglich durch Rechtsklick auf das gewünschte Bild die Adresse dazu kopieren. Seid ihr zum Beispiel Fußballfan und ihr wollt das Logo zu eurem Lieblingsverein zu der Botantwort nutzen, könnt ihr dies tun.



Schaut euch für diese Aufgabe einmal genau in der domain.yml um. Findet ihr die passende Syntax für das Hinzufügen eines Bildes? Falls ja, schreibt sie euch auf.

- 2) Stellt euren Chatbot den anderen vor. Erzählt ihnen am besten nur welches Thema ihr eingepflegt habt. Nicht welche Fragen sie stellen sollen. Tauscht untereinander die Plätze und schaut, ob ihr herausfindet was verändert wurde. Bitte achtet dabei auf den Mindestabstand.
- 3) Möchtet ihr lieber noch etwas zum Thema Pokémon hinzufügen? Dann erweitert die Pokemonliste mit euren Lieblingspokemon aus anderen Staffeln. Fügt dazu in der Datei „pokenames.txt“ im Ordner „data“ weitere Pokemon hinzu und speichert die Datei. Erkennt der Bot nach erneutem Training euer neues Pokemon?

Tipp: Falls er das Pokémon nicht erkennt, fügt es in eine Frage unter dem Intent „confirm_exist“ in der nlu.yml ein.

```
- intent: confirm_exist
  examples: |
    - Ist [Bisasam](pokemon_name) ein Pokemon?
    - Kennst du [Piepi](pokemon_name)?
```

Beispiel mit Wailord:

```
- intent: confirm_exist
  examples: |
    - Ist [Bisasam](pokemon_name) ein Pokemon?
    - Ist [Wailord](pokemon_name) ein Pokemon?
```

Fügt euer Pokémon am besten noch in die „lookup-Liste“ am Ende der nlu.yml ein.