

Künstliche Interaktionspartner*innen an Hochschulen

Michael Raunig



Michael Raunig

**Künstliche Interaktionspartner*innen
an Hochschulen**

Impressum

Künstliche Interaktionspartner*innen an Hochschulen

herausgegeben vom Verein Forum Neue Medien in der Lehre Austria
Graz, 2020

Autor

Michael Raunig

ISBN

9783751998925

9783200071346 (EPUB)

Druck und Verlag

Books on Demand GmbH, Norderstedt

Inhalt

Vorwort	7
1 Einleitung	9
1.1 Chatbots an Hochschulen?.....	9
1.2 Die Arbeitsgruppe	9
1.3 Zum Leitfaden	11
1.4 Künstliche Interaktionspartner*innen.....	12
2 Einsatzideen und -szenarien für KIP an Hochschulen	17
2.1 Theoretische Verortung von KIP.....	17
2.2 Lehr-/Lern-Settings	18
2.3 Organisatorische Settings	23
3 Technische Lösungen und Werkzeuge	27
3.1 Grundlegende Konzepte und Begriffe	27
3.2 Konkrete Werkzeuge	33
4 Ausblick	71
4.1 Vorteile und Potenziale	71
4.2 Probleme und Herausforderungen	72
4.3 Lessons für die Praxis.....	77
5 Literaturverzeichnis	81

Vorwort

„Der Verein Forum Neue Medien in der Lehre Austria (fnma) bietet als etablierte Interessenvertretung im tertiären Bildungssektor ein lebendiges Netzwerk für die Entwicklung und Umsetzung von Strategien und Maßnahmen im Bereich der digitalen Bildungsangebote.“

Basierend auf seinem Mission Statement fördert der Verein bereits seit vielen Jahren Projekte seiner Mitglieder, die sich mit neuen Aspekten des Einsatzes von Technologien in der Hochschullehre befassen. Im Fokus stehen dabei stets die Entwicklung und der Einsatz dieser in Verbindung mit Community-bildenden Maßnahmen oder mit einem Forschungsvorhaben. Die Verwertbarkeit der Projektergebnisse durch möglichst viele fnma-Mitglieder – zum Beispiel in Form einer frei zugänglichen Publikation – ist dabei das Hauptziel der Fördermaßnahme.

Die vorliegende Publikation versteht sich als Leitfaden im Bereich der künstlichen Interaktionspartner*innen (Chatbots) an Hochschulen. Dabei werden zunächst Einsatzideen und -szenarien in Form von Lehr-/Lernsettings sowie organisatorischen Settings skizziert. Den Kern der Publikation bildet die Beschreibung von technischen Lösungen und Werkzeugen. Hier werden grundlegende Konzepte und Begriffe erklärt und danach konkrete technische Lösungen dargestellt, wobei auf unterschiedliche Einsatzmöglichkeiten eingegangen wird. Im abschließenden Ausblick werden Vorteile und zukünftige Herausforderungen thematisiert und Tipps für den praktischen Einstieg in das Thema gegeben.

Unter der Leitung des Zentrums für digitales Lehren und Lernen der Universität Graz waren insgesamt acht österreichische Hochschulen am Projekt beteiligt. Der Verein freut sich über diese weitreichende hochschulübergreifende Zusammenarbeit und bedankt sich bei allen Mitwirkenden für ihr großes Engagement im Projekt, ohne das diese Publikation nicht hätte zustande kommen können.

Ihnen, liebe Leserinnen und Leser, wünschen wir eine interessante Lektüre und hoffen, dass Sie durch die Publikation angeregt werden, auch selbst einmal ein Förderprojekt beim Verein einzureichen. Sie finden die zugehörigen Ausschreibungen als Ankündigung im Newsletter oder auf der Webseite <https://www.fnma.at>.

Martin Ebner & Hans-Peter Steinbacher
Präsidenten des Vereins „Forum Neue Medien in der Lehre Austria“

1 Einleitung

1.1 Chatbots an Hochschulen?

In den letzten Jahren hat sich der Trend zum Einsatz von Chatbots und digitalen Assistent*innen (eine begriffliche Klärung folgt unten, Abschnitt 1.4) laufend verstärkt und in weiten Bereichen des täglichen Lebens – im beruflichen und kommerziellen, aber auch im sozialen und privaten Umfeld – etabliert. Die Unterhaltung mit künstlichen Gesprächspartner*innen gehört zunehmend zum Alltag. Die technologischen Fortschritte in den Bereichen *Künstliche Intelligenz* und *Machine Learning* sowie die voranschreitende Digitalisierung (als gesellschaftlicher Prozess) schüren Hoffnungen und Ängste gleichermaßen, oftmals verbunden mit unrealistischen Vorstellungen dieser Phänomene. Die verschiedenen Einsatzmöglichkeiten für künstliche Gesprächspartner*innen werden jedenfalls zahlreicher und bilden auch für Hochschulen eine Inspirationsquelle: Der *Horizon Report 2020* (Teaching and Learning Edition) etwa nennt (wie schon in den Vorjahren) „AI/Machine Learning Education Applications“ als eine der maßgeblichen lehr-/lerntechnologischen Trends für den Hochschulbereich und führt Chatbot-Services an Hochschulen als herausragendes Beispiel dafür an (Brown et al., 2020, S. 16ff.).

1.2 Die Arbeitsgruppe

Anlässlich dieser Entwicklungen und Aussichten haben wir uns entschieden, uns intensiver mit Bots und Assistent*innen in Form eines Projekts zu befassen. Grundidee war es, sich mit bestehenden Technologien (Bot-Plattformen, digitalen Assistent*innen, KI-Systemen und -Anwendungen) explorativ auseinanderzusetzen und erste Kompetenzen in diesem relativ neuen Feld aufzubauen – nicht nur Kenntnisse im Bereich der relevanten Technologien und Anwendungen, sondern auch ein besseres Verständnis der dadurch eröffneten Einsatz- und Interaktionsmöglichkeiten sowie Perspektiven für den Hochschulbereich. Um hochschulüber-

greifend Expertise und Know-how zu bündeln, sollte eine größere Arbeitsgruppe eingerichtet werden, die Erkenntnisse und Ressourcen zum Thema auf einer breiten Basis sammelt, diese auf einer öffentlich zugänglichen Webplattform dokumentiert sowie schließlich in Form einer Publikation veröffentlicht. Für diese Vorgehensweise sprachen der relativ junge und wenig überschaubare Markt an technologischen Lösungen, die eher spärliche Literatur (und nur in wenigen Fällen intensivere Auseinandersetzung mit dem Thema) und die über verschiedene Hochschulen und Hochschultypen hinweg ähnlichen Bedarfe und Herausforderungen, denen man mit künstlichen Gesprächspartner*innen begegnen kann. Bei der Sondierung und Erprobung stellte es sich aber auch als hilfreich heraus, unterschiedliche Ausgangssituationen, Anforderungen und Bedarfe gleich von Anfang an zu berücksichtigen.

Die Ausschreibung der Projektförderung des Vereins „Forum Neue Medien in der Lehre Austria“ (fnma)¹ im Jahr 2019 bot die Gelegenheit, dieses Vorhaben österreichweit in die Tat umzusetzen. Neben weiteren Projekten erhielt der von der Universität Graz eingebrachte Antrag den Zuschlag, und die Arbeitsgruppe konnte im November 2019 die Arbeit aufnehmen.

Ich darf mich als Projektleiter an dieser Stelle bei fnma für die unbürokratische Förderung und freundliche Unterstützung sowie bei den Arbeitsgruppenmitgliedern für die Mitwirkung an den Arbeitsgruppentreffen, an der Recherche und Materialsammlung, an Diskussionen und schließlich für das Feedback zum Leitfaden bedanken:

- Franziska Chen
- Markus Ebner
- Martina Friesenbichler
- Elke Höfler
- Ursula Leopold
- Christina Mossböck

¹ <https://www.fnma.at/projekte/aktuelle-ausschreibung>

- Markus Orthaber
- Natasa Sfiri
- Josef Smolle
- Daniela Wolf
- Eva Zöserl

Das im Rahmen der Arbeitsgruppe entwickelte Informationsangebot umfasst eine kurze Projekt-Homepage² mit den wichtigsten Informationen sowie eine Dokumentationsplattform, die der Sammlung und steckbriefartigen Dokumentation dient und in Form eines Github-Wikis realisiert wurde.³ Das zentrale Ergebnis der Auseinandersetzung ist jedoch der vorliegende Leitfaden.

1.3 Zum Leitfaden

Die Intention des Leitfadens ist es, eine aktuelle Bestandsaufnahme zu liefern und folgende Fragen zu beantworten: Wozu sollen und wie können Bots an Hochschulen erstellt und eingesetzt werden? Der Leitfaden ist generell aus der Hochschulperspektive geschrieben – er adressiert somit in erster Linie hochschulspezifische Fragen und Problemstellungen (die sowohl die Lehre, Organisation als auch den Ressourcenhorizont betreffen). Zudem liefert der Leitfaden einen praxisorientierten Überblick, die wissenschaftliche Auseinandersetzung steht – wenngleich in einigen Teilen angelegt – nicht im Vordergrund; auch die genauen Details zu einzelnen Szenarien oder Werkzeugen (im Sinne einer umfassenden und erschöpfenden Darstellung) werden zugunsten einer exemplarischen bzw. illustrativen Darstellung ausgespart. Die Informationen zu den einzelnen Werkzeugen und Services wurden im Rahmen von praktischen Tests gewonnen und durch die Lektüre von begleitenden Materialien (Dokumentation, Anleitungen etc.) ergänzt. Die Szenarien und abschließenden Erkenntnisse entstammen einer Literatur- und Web-Recherche

² <http://kiph.fnma.at/>

³ <https://github.com/erematanmw/kiph/wiki>

sowie teilweise auch eigenen Überlegungen. Es wird keinerlei Anspruch auf Vollständigkeit und – aufgrund der sich schnell wandelnden Angebote – auch kein Anspruch auf Aktualität in den nachfolgenden Ausführungen erhoben.

Formal bedient sich der Leitfaden der Gender-Sternschreibung – um einerseits der Diversität menschlicher Akteur*innen sprachlich gerecht zu werden und andererseits Unklarheiten und Ungerechtigkeiten in der Geschlechtszuweisung für künstliche Systeme zu umgehen.

Ansonsten steht der Leitfaden unter der Lizenz CC BY 4.0⁴, darf also unter Nennung des Verfassers beliebig vervielfältigt, verbreitet und verändert werden.

1.4 Künstliche Interaktionspartner*innen

Die Bezeichnung *Künstliche Interaktionspartner*in* umfasst im vorliegenden Leitfaden sowohl Chatbots als auch digitale Assistent*innen und verkörperte Gesprächsroboter.

Chatbots sind spezielle Programme, deren Zweck bzw. Aufgabe in der autonomen Kommunikation mit einem (menschlichen) Gegenüber besteht. Meist wird der Begriff *Chatbot* im Zusammenhang mit speziellen Kommunikationszwecken gebraucht (z. B. Chatbots für die Bereitstellung von Informationen oder die Bestellung eines Produkts), während mit *Social Bots* weitgehend „virtuelle Persönlichkeiten“ gemeint sind – also Gesprächspartner*innen, die einen eigenen Charakter haben und wie „echte“ Personen agieren.

Im Gegensatz zu Chatbots, mit denen man sich üblicherweise durch Eingabe von Text und Betätigung von Buttons auf einer Weboberfläche oder in einer App unterhält, ist es durch die Verbindung mit geeigneten Ein-/Ausgabegeräten und Spracherkennung bzw. Sprachausgabe möglich, mit sogenannten *digitalen Assistent*innen* auch in gesprochener Sprache zu interagieren. Bekannte Beispiele für

⁴ <https://creativecommons.org/licenses/by/4.0/deed.de>

digitale Assistent*innen (oder Sprachassistent*innen, oder etwas unpersönlicher: Voice User Interfaces) sind etwa Alexa (Amazon), Siri (Apple), der Google Assistant oder Cortana (Microsoft). Die Unterhaltung kann auf diversen digitalen Endgeräten erfolgen – mittels eines herkömmlichen PCs mit Mikrofon und Lautsprecher, auf einem Smartphone oder mit speziellen Devices (*sprachgesteuerte Lautsprecher* oder *Smart Speaker* wie z. B. Amazons Echo oder Googles HomePod).

Weniger verbreitet und (zumindest in Europa) noch relativ neu sind sprachbegabte Roboter, die ebenfalls als (wenn auch exotische) künstliche Interaktionspartner*innen mit spezifischer Charakteristik (Verkörperung, zusätzliche non-verbale Kommunikationsfähigkeiten, Ortsgebundenheit) gelten können und in einem kurzen Abschnitt (unter 3.3.8) behandelt werden.

Zur Berücksichtigung sämtlicher unterschiedlicher (künstlicher) Systeme wurde bewusst der breite Begriff *Interaktionspartner*in* verwendet; damit sollten auch weitere Arten der hochschulrelevanten Interaktion neben der Kommunikation miteinbezogen oder zumindest mitgedacht werden. Ohne eine bestimmte Theorie der Interaktion vorauszusetzen, soll die Auseinandersetzung mit künstlichen Interaktionspartner*innen – im Folgenden abgekürzt als KIP – nicht nur dialogische Unterhaltungen fokussieren, sondern auch umfassendere (wenn auch bislang kaum realisierte) Settings mitdenken, etwa die Automatisierung von Teilen didaktischer Praktiken, neue Formen der multimodalen Instruktion etc. Alternative in Frage kommende Begriffe (wie etwa *Agent*in* oder *Kommunikations-/Interaktions-Interface*) erschienen zu allgemein oder tendenziell auch mit Konnotationen behaftet.

Die Auseinandersetzung mit Chatbots und Co. mag auf den ersten Blick vielleicht als Spielerei wirken – tatsächlich können KIPs aber auf eine umfangreiche Vorgeschichte in Literatur und Film zurückblicken, und die ersten seriösen Versuche reichen bis in die Mitte des vorigen Jahrhunderts zurück (Joseph Weizenbaums Programm ELIZA wird häufig als erstes Beispiel genannt). Bisweilen werden Chatbots sogar als die nächste Entwicklungsstufe der web- und internetbasierten

Anwendungen gesehen, die Apps ersetzen würden und diese als User Interface in mancher Hinsicht überlegen seien („Bots are the new apps“).

Wie auch immer man den Gehalt solcher Einschätzungen bewertet – jedenfalls scheint vorab eine Einteilung von KIPs hilfreich:

- Als Minimalvariante können automatisierte Kommunikationsagent*innen gelten, die kaum Interaktion zulassen (z. B. skript-basierte Bots, die vorgefertigte oder dynamisch generierte Inhalte zu bestimmten Zeiten oder Anlässen vermelden oder anhand von simplen Regeln „interagieren“, etwa einfache Antworten geben).
- Etwas differenzierter agieren Korrespondenz-Bots, die auf vordefinierte Eingaben oder Muster antworten oder nach bestimmten Spielregeln und Verlaufsplänen funktionieren; darunter fallen z. B. einfache Kommunikations-Bots für Messenger-Dienste.
- Die gegenwärtige Maximalvariante – sprich: mit den aktuell verfügbaren Technologien und Angeboten sowie mit überschaubaren Mitteln einigermaßen realisierbar – sind Gesprächspartner*innen, deren Verständnisfähigkeit nicht nur mit einer Vielfalt von möglichen Ausdrucksweisen und Gesprächsfortgängen zurande kommt, sondern die auch Kontextinformationen ins Gespräch miteinbeziehen und adaptiv reagieren können. „Intelligente“ oder „smarte“ Bots aus dem Showcase einschlägiger bzw. potenter Firmen (etwa Facebooks Blender⁵, Googles Meena⁶ oder auch Klassiker wie etwa Mitsuku⁷) stehen zwar am Ende der Evolution von *conversational artificial intelligence*, sind aber nicht für den breiten Einsatz gedacht – und meist auch gar nicht erforderlich.

⁵ <https://ai.facebook.com/blog/state-of-the-art-open-source-chatbot/>

⁶ <https://ai.googleblog.com/2020/01/towards-conversational-agent-that-can.html>

⁷ <http://www.mitsuku.com/>

Vor einer detaillierteren Darstellung weiterer Begriffe, technischer Aspekte und Lösungen für KIPs werden deshalb zunächst Einsatzideen und -szenarien an Hochschulen geschildert, um den Rahmen aus der Einsatzperspektive abzustecken und eine erste praktische Orientierung zu ermöglichen. Obwohl das Verhältnis von Technik und Einsatz mittlerweile differenzierter betrachtet wird (vgl. beispielsweise Krommer, 2018), scheint es hier sinnvoll, zunächst den Zweck vor den einzelnen Werkzeugen zu betrachten und sich zuerst darüber klar zu werden, wie der Einsatz an Hochschulen aussehen könnte.

2 Einsatzideen und -szenarien für KIP an Hochschulen

2.1 Theoretische Verortung von KIP

Zur Einstimmung auf konkrete Szenarien folgt zunächst ein kurzer Abriss verschiedener didaktisch-theoretischer Begriffe und Konzepte, in deren Zusammenhang Bots eine Rolle spielen (können). Wie auch bei der Schilderung einzelner Settings (Abschnitt 2.2) steht dabei nicht eine fundierte Auseinandersetzung im Vordergrund; vielmehr sollen praktische Einsatzmöglichkeiten und mögliche Ansatzpunkte für eine vertiefende Betrachtung aufgezeigt werden.

Generell können KIPs als neue Form bzw. Weiterentwicklung von „pedagogical agents and traditional intelligent tutoring systems“ (Winkler & Söllner, 2018, S. 3) gelten, also als zeitgemäße Ausprägung des Versuchs, elektronische/digitale Lernsysteme bzw. deren Interfaces „menschlicher“ oder menschenähnlicher zu gestalten. Dabei gibt es Ansätze, die KIPs als mögliche Ergänzung von bestehenden Lernmanagementsystemen (LMS) und Systemen zur Dokumentation des Lernfortschritts mit dem Ziel eines umfassenden „elektronischen Performance Support Systems“ (vgl. Hahn, 2019, Kap. 6) betrachten; andernorts werden KIPs zu „unsichtbaren“ Lern- bzw. Wissensmanagementsystemen erklärt, wobei die Funktion eines solchen „invisible LMS“ (Clark, 2017) weniger in der Ergänzung, sondern im Ersatz für (mitunter als unattraktiv oder unpraktisch empfundene) Lernmanagementsysteme liegt. Besonders im Training und in der praktischen Ausbildung erweisen sich KIPs als geeignetes Instrument für das „Learning in the Workflow“ (vgl. z. B. McElvaney, 2018), indem sie als Lernbegleiter*innen direkt in den Lernprozess eingebettet sind.

Im Zusammenhang mit KIPs kommen teilweise auch neue Begriffe und Bezeichnungen für Formen des Lernens auf. So bilden KIPs etwa einen wesentlichen Fak-

tor im Trend zum „assisted learning“ (mmb Institut, 2019) oder ermöglichen „conversational learning“ (vgl. Kowald & Bruns, 2019) – also begleitete und in Form von Gesprächen und Unterhaltungen angestoßene Lernprozesse, die ja hinlänglich von und mit Menschen praktiziert werden, neuerdings aber auch durch digitale Technologien realisiert werden können.

Neben lerntheoretischen Aspekten bilden vor allem „intelligenter“ KIPs auch ein Teilgebiet des Einsatzes von Künstlicher Intelligenz in der Hochschullehre (zu deren vielfältigem Spektrum vgl. ausführlich Zawacki-Richter et al., 2019).

2.2 Lehr-/Lern-Settings

2.2.1 Fachliche Lernunterstützung

Eine erste Möglichkeit des didaktischen Einsatzes von KIPs besteht darin, fach- oder lehrveranstaltungsspezifische Lernprozesse zu unterstützen. Dies kann beispielsweise in Form eines strukturierten reflexiven Gesprächs erfolgen, mit dem Ziel, die Reflexion über ein bestimmtes Themengebiet oder über eine praktische Lernaufgabe zu fördern und generell Reflexion als (Lern-)Kompetenz zu erlernen. Beispiele sind etwa der „confessional chatbot Beau“ (beschrieben in den Hollander et al., 2019) oder „Rebo“, der am Grazer Know-Center für die Lehrlingsausbildung entwickelt wird und Lehrlinge mit Reflexionsfragen, periodischen Zusammenfassungen und Wiederholungen unterstützt – ein Setting, das auch auf Lehrveranstaltungen an Hochschulen übertragbar ist, wie das an der Hochschule Ruhr West angebotene „virtuelle Lehrgespräch“ beweist. Zur Festigung des Lernstoffs einer Vorlesung lässt sich dort ein Chatbot von Studierenden die Kernaussagen der vergangenen Vorlesung erläutern, stellt Fragen und geht auf die Antworten der Nutzer*innen ein.⁸

⁸ Für Außenstehende ist es üblicherweise schwierig, Einblick in diese nicht-öffentlichen (da prototypisch und experimentell angelegten) Entwicklungen und Produkte nehmen zu können. Meist muss man auf Pressemitteilungen – im Fall des virtuellen Lehrgesprächs

Neben der Reflexion bieten sich einige Fertigkeiten und Kompetenzen an, die besonders gut von KIPs vermittelt und mit ihnen trainiert werden können – allen voran das Lernen von Sprachen. Im kommerziellen Bereich illustrieren verschiedene Anbieter (z. B. Magiclingua oder Duolingo, wo man bis vor einiger Zeit auch mit Sprachbots üben konnte) die Möglichkeiten von digital realisiertem Sprachtraining. Die Umsetzungsmöglichkeiten von Bots beim Studium und Erlernen von Sprachen sind aber vielfältig – man denke etwa an die Transformation bestehender E-Learning-Angebote in Bot-Services. Aber auch nicht direkt auf Sprache bezogene, sondern auf der Gesprächsform beruhende Settings können mit KIP realisiert werden, etwa die Simulation von Gesprächsverläufen z. B. in der Therapieausbildung, wo KIPs als Simulationspatient*innen bzw. Klient*innen dienen und beim Ausbilden der eigenen therapeutischen Fertigkeiten helfen können.

2.2.2 Begleitung und Strukturierung von Lehrveranstaltungen

Weniger für die inhaltliche Gestaltung, aber zur organisatorischen Unterstützung von Lehrveranstaltungen dienen digitale Terminassistent*innen: Anstelle von Kalenderfunktionen von Groupware oder im Campus- oder Lernmanagementsystem können Studierende von KIPs an Termine, Auf- und Abgaben, geplante Aktivitäten etc. erinnert bzw. in Form eines Dialogs zum Aktivwerden angeleitet werden; ein solches Service kann entweder individuell als einfacher Bot (beispielsweise in Slack) oder hochschulweit als KIP mit Anbindung an zentrale Verwaltungssysteme realisiert sein.

Inhaltlich kompetenter (und auch komplizierter zu realisieren) können künstliche Tutor*innen bzw. Studienassistent*innen sein, die Lehrende während einer Lehrveranstaltung unterstützen bzw. in Bezug auf automatisierbare Aufgaben entlasten, etwa allgemeine Hilfestellungen bieten, wiederkehrende Fragen beantworten etc.

etwa <https://nachrichten.idw-online.de/2020/01/30/ausgezeichnete-innovative-lehre-virtuelles-lehrgesprach-mit-dem-chatbot-durchs-studium/> – oder auf Vorträge wie etwa beim Know-Center zurückkommen (<https://www.know-center.tugraz.at/research/publikationen/vortraege/#542>).

Ein frühes, elaboriertes und recht bekannt gewordenes Beispiel ist etwa eine virtuelle Lehrassistentin am Georgia Institute of Technology namens „Jill Watson“, die Studierendensupport für eine Lehrveranstaltung zusammen mit menschlichen Kolleg*innen leistet(e) und angeblich nicht von diesen unterschieden werden konnte (vgl. Goel, 2016). Ein Pionier beim Einsatz von physischen Robotern ist Jürgen Handke, der in verschiedenen Projekten (z. B. H.E.A.R.T. – Humanoid Emotional Assistant Robots in Teaching⁹) Roboter im Hochschulalltag bzw. in der Hochschullehre an der Universität Marburg erprobt und untersucht hat (zu möglichen Settings vgl. ausführlich Handke, 2020, Kap. IV).

2.2.3 Punktueller Einsatz als Lehr-Tool

Wenn die künstliche Betreuung nicht über die gesamte Lehrveranstaltung oder über das Semester hinweg erfolgen soll, bietet sich der punktuelle Einsatz von KIPs für bestimmte Phasen oder Aktivitäten an. Dies kann beispielsweise zu Beginn einer Lehrveranstaltung in Form einer Erstororientierung oder eines „Onboardings“ geschehen, wo ein*e KIP mittels strukturierter Gesprächsverläufe allgemeine Informationen und Voraussetzungen kommunizieren, Vorkenntnisse, Erwartungen und Wünsche abfragen sowie kursspezifische Fragen (Beurteilungskriterien, Prüfungsmodalitäten etc.) beantworten kann.

In diskurs- und interaktionsorientierten Settings können KIPs für die Vermittlung und Vernetzung von Personen ihren Einsatz finden, indem etwa Terminfindungs-, Gruppenbildungs- und Abstimmungsprozesse mithilfe von Teamorganisations-Bots unterstützt werden; diese fungieren dann als vermittelnde Zwischenglieder in der Kommunikation sowohl zwischen Lehrenden und Studierenden als auch zwischen Studierenden untereinander und können solche Prozesse strukturieren und „moderieren“.

In Lehrveranstaltungen können weiters formative Assessments in Form von Bot-Dialogen durchgeführt werden, anstatt herkömmliche Selbsttest-Lösungen und -

⁹ <https://www.project-heart.de/>

Umgebungen zu bemühen. In ähnlicher Form kann auch bei der Lehrveranstaltungsevaluation auf eine*n Umfrage-KIP anstelle von Umfragen in Form eines Formulars o. Ä. zurückgegriffen werden (vgl. beispielsweise „Hubert“, der später vorgestellt wird). Schließlich kann man auch die Studierenden selbst einen Bot als Projekt- oder Abschlussarbeit entwickeln lassen und diese Leistung zur Beurteilung heranziehen, wie das mitunter bereits mit der Erstellung von Videos oder Lernobjekten (anstelle von Dokumenten und Klausuren) gehandhabt wird. Leistungen in KIP-Form scheinen insbesondere geeignet bei solchen Inhalten und Themen, die dialogisch oder im Frage-Antwort-Format vermittelt werden können.

2.2.4 Individuelle Betreuung von Lernenden

Das Betreuungsangebot, das Hochschulen für ihre Studierenden zur Verfügung stellen, reicht üblicherweise über einzelne Lehrveranstaltungen hinaus und wird auch im Sinne einer zeitgemäßen Hochschulentwicklung laufend an neue Anforderungen und Formate angepasst. KIPs könnten beispielsweise Lernkompetenzen fördern, indem sie als Coach-Bots zur Steigerung von Lerneffizienz und Produktivität, für die Verbesserung des eigenen Selbstmanagements, zur Vermeidung von Prokrastination u. Ä. eingesetzt werden (vgl. etwa Grams, 2019 zum Potenzial von Bots im professionellen Coaching-Bereich). Vor allem im amerikanischen Hochschulraum verbreitet und für einige fachliche Disziplinen relevant ist das Mentoring, also die persönliche Begleitung und Förderung von Studierenden durch erfahrene, fortgeschrittene Kolleg*innen. Obwohl – noch mehr als beim Coaching – die persönliche Beziehung zwischen Mentor*in und Mentee im Vordergrund steht, ließen sich auch hier „skalierbare“ Mentoring-Angebote mittels „erfahrener“ KIPs realisieren – wenngleich hier wohl ein intensives Training der Mentor*innen-KIPs und mehr Augenmerk auf deren „Persönlichkeit“ gelegt werden müsste als etwa beim Coaching. Ein erster Schritt in diese Richtung wären KIPs als Mittel, um adaptive Lernpfade zu vermitteln und personalisiertes Lernen zu ermöglichen, indem KIPs den individuellen Lernbedarf im Gespräch eruieren und mit entsprechenden Angeboten aufwarten (vgl. White, 2019).

Weitaus bescheidenere Anforderungen stellen Motivations-Bots, die Studierende mit Lob und Ansporn dabei helfen, ihre Lernziele zu erreichen. Neben der Förderung von „student retention“ und der Vermeidung von Drop-out können Motivations-KIPs helfen, die typischen Aktivitätskurven in Lehrveranstaltungen oder während des Semesters abzufedern und auszugleichen. Eine einfache Umsetzung von Motivations-KIPs könnte in Form eines Moodle-Plugins erfolgen (vgl. Wittke, 2017); in ambitionierten Vorhaben könnten solche KIPs auch mit Learning Analytics und/oder Gamification-Ansätzen kombiniert werden. Mittels Tracking des Lernfortschritts, des Vergleichs mit anderen Lernenden und spielerischen Elementen kann der Lernprozess abwechslungsreicher gestaltet, mit Herausforderungen versehen werden und Wettbewerbscharakter annehmen (im positiven Sinn von Lernanreizen bzw. des gezielten persönlichen Leistungsmonitorings). Damit verbunden ist auch die Idee, eine*n KIP als Alternative zu einem webbasierten Lernenden-Dashboard im Rahmen von Learning-Analytics-Systemen anzubieten und damit die Ergebnisse für die Lernenden ansprechender aufzubereiten.¹⁰

Durch ihre unterschiedlichen Ausformungen und jeweilige Charakteristik bieten KIPs ganz generell vielfältige Unterstützungsmöglichkeiten für Studierende mit Beeinträchtigungen. Unterschiedlich gestaltete KIPs können Studierenden mit körperlichen oder psychischen Beeinträchtigungen alternative Interaktions- und Vermittlungsmedien im Hochschulalltag bereitstellen und einen Baustein im Bemühen bilden, ein „universal design“ in der Lehre, aber auch im allgemeinen Hochschulbetrieb zu erreichen. Darüber hinaus eignen sich KIPs prinzipiell auch für die (allgemeine) psychologische Betreuung von Studierenden – das belegen nicht nur die Erfolge künstlich bewerkstelligter Verhaltenstherapie (vgl. etwa den Woebot¹¹ als

¹⁰ Bots können in der Regel relativ einfach mit anderen Systemen verknüpft werden, was aber andere Probleme (Datenschutz etc.) mit sich bringt, auf die weiter unten eingegangen wird; die geschilderte Bot-Vision ist außerdem abhängig vom Fortschritt und Implementierungsstatus von Learning Analytics, das hierzulande noch in den Kinderschuhen steckt.

¹¹ <https://woebot.io>

prominentes Beispiel), sondern auch die in Niebuhr (2018) beschriebenen Ansätze gegen hochschulischen Lern- und Prüfungsstress.

2.3 Organisatorische Settings

Abseits vom Lehrbetrieb gibt es weitere Formen der Kommunikation von Hochschulen mit ihren Bediensteten und Studierenden sowie weiteren Zielgruppen (Studieninteressierten, Alumni, Öffentlichkeit). Allgemeine Informationen werden üblicherweise in einem Campusmanagementsystem vorgehalten, wo man auf einer Web-Oberfläche Informationen zu Lehrveranstaltungen, Studien, Räumen, Personen oder Veranstaltungen erhalten und auch administrative Schritte (Anmeldung zu Veranstaltungen, Beantragung von Services etc.) setzen kann. Ein*e Campusmanagement-KIP als hochschulweite Lösung könnte diesbezügliche Auskünfte erteilen; hier setzt z. B. der Searchchatbot der TU Graz an (vgl. Berger et al., 2019). Nach erfolgter Authentifizierung könnten KIPs auch die Anmeldung zu Veranstaltungen ermöglichen, an Termine erinnern, Prüfungsergebnisse kommunizieren etc. – in diese Richtung geht etwa „Charly – Der digitale Assistent“¹² (ein kommerzielles Angebot), der an das in Österreich weit verbreitete Campusmanagementsystem „CAMPUSonline“ angebunden ist. Ein ähnliches Vorhaben verfolgt das Projekt „Mobile First for Students“¹³ im Rahmen der BMBWF-Ausschreibung „Digitale und soziale Transformation in der Hochschulbildung“¹⁴, in welchem die umfassende Nutzung von Campus-Services in einer Smartphone-App mit integrierten Chatbots und Sprachdiensten unterstützt werden soll.

Im US-amerikanischen Raum sind bereits – so entnimmt man zumindest den Informationsangeboten – kommerzielle Systeme für die Begleitung und Förderung

¹² <https://www.indib.it/hochschulen/charly-der-digitale-assistent/>

¹³ <https://www.wu.ac.at/universitaet/news-und-events/news/details-news/detail/2-digitalisierungs-projekte-gefoerdert>

¹⁴ <https://www.bmbwf.gv.at/Ministerium/Presse/Digitale-soziale-Transformation-HS.html>

von Studierenden im kompletten „academic life cycle“ etabliert (z. B. Ivy¹⁵ oder AdmitHub¹⁶), welche Studierenden von der Studiumszulassung bis hin zum Berufseinstieg virtuelle Assistent*innen zur Seite stellen; als Nebenprodukt dieser digitalen Ausprägung der Servicierungskultur an amerikanischen Hochschulen fallen freilich auch Daten an, die für die Optimierung von Angeboten sowie Learning Analytics genutzt werden können.

Weniger ambitioniert und einfacher realisierbar (und hierzulande datenschutzrechtlich weniger brisant) sind basale Unterstützungsangebote in Form von KIPs, die z. B. einzelne Schritte im Studium begleiten, eine Entlastung des Beratungspersonals erreichen und ein weiteres „Informationsformat“ darstellen sollen. Einsatzgebiete sind z. B. die Studienwahl und Studienorientierung an einer Hochschule (zur Implementierung vgl. beispielsweise Carstensen & Roedenbeck, 2019), die Studienberatung bzw. konkrete Beantwortung von Fragen zum Studium (wie etwa „Leo“, der Chatbot der Universität Innsbruck, der Fragen rund ums Studium auf der Webseite¹⁷ beantwortet), oder Fragen im Zusammenhang mit Aufnahme- und Zulassungsverfahren etc. Aber auch Auskunft und Support für rein technische oder administrative Fragestellungen werden vermehrt mithilfe von KIPs realisiert – so beantwortet etwa ein Chatbot der Universität Wien Fragen rund um den Ausweis („u:card“¹⁸), während der digitale Assistent „ASK“ der Universität Graz¹⁹ Auskünfte zur IT-Infrastruktur und den verschiedenen IT-Services bietet. (Da den Hochschulbediensteten oft interne Informationen zur Verfügung gestellt werden, sind deren Assistent*innen nur nach Authentifizierung oder im Hochschul-Intranet erreichbar.)

¹⁵ <https://ivy.ai>

¹⁶ <https://www.admithub.com>

¹⁷ <https://www.uibk.ac.at/studium/>

¹⁸ <https://ucard.univie.ac.at/studierende/>

¹⁹ <https://it.uni-graz.at/de/digitaler-assistent/>

Mittels KIPs wäre zudem etwa ein Campus-Guide-Service realisierbar, das Orientierung und ortskundige Auskunft am Campus bietet – primär nicht in Form eines Chatbots im Web, sondern als ortssensible Auskunfts-App oder ortsgebundenes Service in Form von (vor Ort installierten) Sprachassistent*innen oder Robotern. („Empfangsroboter“ sind vereinzelt schon bei kommerziellen Unternehmen anzutreffen.)

Klassische Chatbots wiederum sind schließlich als Werkzeug des „Brand Managements“ von Hochschulen (neben der Präsenz auf sozialen Netzwerken) geeignet, die mit ihren Zielgruppen auch auf diese Weise „in den Dialog treten“ können. Auf Wunsch könnte ein Hochschul-Chatbot auf populären Messenger-Plattformen Hochschul-Smalltalk führen, Marketingaktionen starten und „lead generation“ von Studien- bzw. generell an den Aktivitäten der Hochschule Interessierten betreiben.

In den genannten Einsatzbeispielen klingen bereits unterschiedliche Voraussetzungen und Umsetzungsmöglichkeiten an – von komplexen technischen Eigenentwicklungen und der Anmietung kommerzieller Systeme bis zur Umsetzung mithilfe von universellen oder auf bestimmte Produkte zugeschnittenen Chatbot-Baukästen. Im folgenden Abschnitt werden die aktuellen Optionen zur Erstellung von künstlichen Interaktionspartner*innen systematisch dargestellt; der Fokus liegt dabei auf – auch für Nicht-Programmierer*innen – handhabbaren Lösungen, mit denen man brauchbare Ergebnisse mit überschaubarem Budget und Aufwand erzielen kann.

3 Technische Lösungen und Werkzeuge

3.1 Grundlegende Konzepte und Begriffe

3.1.1 Interaktionsplattformen

Obwohl es der Begriff suggeriert, sind KIPs bzw. Bots keine homogenen, klar abgegrenzten technischen Gebilde bzw. Einheiten, sondern in der Regel aus verschiedenen Softwarekomponenten und Modulen zusammengesetzte Systeme, die jedoch ein abgrenzbares und benennbares „Interface“ für die Interaktion mit Menschen bieten. Neben den einzelnen Komponenten, die u. a. für das Verstehen von Eingaben (Text und Sprache), das lexikalische und stilistische Repertoire von KIPs, die Abfrage und Einbindung externer Daten etc. zuständig sind, ist in Hinblick auf die „User Experience“ vor allem der Kanal relevant, auf dem die Interaktion zwischen KIP und Mensch vonstatten geht. KIPs operieren – bis auf physische Geräte wie Smart Speaker und Roboter – nicht im luftleeren Raum, sondern auf bestimmten Umgebungen und Plattformen: Sie können in das jeweilige Betriebssystem integriert sein (wie beispielsweise Siri auf iOS) oder als zusätzliche App installiert werden. Chatbots sind meist in klassischen Messenger- und Kommunikationslösungen (z. B. Facebook Messenger, Line, Skype, Slack, Telegram oder WhatsApp) zu finden oder alternativ als Web-Chat verfügbar, also in eine Webseite eingebettet – meist mit einem Sprechblasen- oder Roboter-Symbol „aktivierbar“ oder als „Widget“ für Web-Contentmanagementsysteme. Relativ neu und noch nicht etabliert ist zudem die Möglichkeit, mit Bots über den Rich-Communication-Services-Standard (RCS)²⁰ zu kommunizieren.

²⁰ Siehe https://de.wikipedia.org/wiki/Rich_Communication_Services. RCS ist quasi die neue Generation von SMS bzw. die Reaktion der Mobilfunkindustrie auf populäre internetbasierte Messenger-Dienste.

Diese einzelnen Kanäle (für die Nutzung von KIPs) werden mitunter auch als „Bot-Plattformen“ bezeichnet – im Gegensatz zu „Bot-Entwicklungsplattformen“, die die Erstellung und Bearbeitung von KIPs (für deren Produzent*innen bzw. Betreiber*innen) ermöglichen.

3.1.2 Verstehen: Intent-Klassifikation

Eine wesentliche Fähigkeit von KIPs besteht darin, die Äußerungen ihres Gegenübers zu „verstehen“. Die Nutzer*inneneingaben und -äußerungen stellen zunächst bloße Zeichenketten oder Lautfolgen dar, die für KIPs zwar bestenfalls unterscheidbar sind, deren Bedeutung jedoch nicht so zugänglich ist wie für Menschen, die semantischen Gehalt verstehen können. Wenn Gesprächsverläufe nicht durch vordefinierte Antwortmöglichkeiten oder Antworten nicht durch allgemeine Regeln (Input-Output-Mappings) bestimmt sind, wird somit ein Mechanismus bzw. eine Abstraktionsebene benötigt, die zwischen der Menge möglicher bzw. sinnvoller Äußerungen und dem, was KIPs verstehen, vermittelt.

Dies kann entweder rein regelbasiert (in einem einfachen Fall etwa: „Wenn in der Äußerung eine bestimmte Zeichenkette vorkommt, dann reagiere mit einer bestimmten Antwort.“) oder mit Machine-Learning-Technologien erfolgen, wo künstliche neuronale Netzwerke „lernen“, auf (auch bislang unbekannte, nicht in den Trainingsdaten vorkommende) Äußerungen mit angemessenen Antworten zu reagieren.

Üblicherweise werden jedoch Äußerungen bestimmten „Intents“ zugeordnet bzw. dahingehend klassifiziert, ob (und in welchem Ausmaß) sie mit den für spezielle KIPs vordefinierten Intents übereinstimmen. Diese Methode wird meist als *Natural Language Understanding* (NLU)²¹ bezeichnet und kann ebenfalls regelbasiert oder

²¹ NLU wird oftmals begrifflich NLP (Natural Language Processing) gegenübergestellt; den Unterschied kann man wie folgt umreißen: „In the most basic terms, NLP looks at what was said, and NLU looks at what was meant.“ (<https://blog.rasa.com/nlp-vs-nlu-whats-the-difference/>)

mit maschinellem Lernen (oder beidem) operieren. Intents sind also Nutzer*innenabsichten hinter den konkreten Äußerungen, auf die KIPs reagieren und für die sie programmiert oder trainiert wurden – beispielsweise einfache Intents wie Begrüßungsformen oder Smalltalk (zur persönlichen Befindlichkeit und über das Wetter), aber auch komplexere Intents wie etwa sich für eine Lehrveranstaltung einzuschreiben oder den Termin einer Veranstaltung zu erfragen. Die Intent-bezogene Konversation gestaltet sich entsprechend den „erlernten“ Vorgaben und kann sich in komplexeren Fällen auch über mehrere Dialogschritte erstrecken und weitere Parameter miteinbeziehen – etwa Objekte bzw. Details von Intents („Entities“), Kontexte und „Sentiments“.

Entities ermöglichen eine kombinatorische Flexibilität und sind gewissermaßen die Variablen von Intents oder Gesprächen; sie werden zu Entitätstypen zusammengefasst und fallen üblicherweise unter folgende drei Arten: eigens definierte (der Klassiker: Pizzasorten; im Hochschulkontext z. B. angebotene Kurse einer Hochschule, Veranstaltungstermine), allgemeine bzw. systembezogene (Datum, Uhrzeit, Länder, Farben etc.) und nutzer*innenspezifische Entities (z. B. Name und Matrikelnummer von Studierenden), die aus externen Datenquellen abgerufen werden können. KIPs, die kontextsensitiv reagieren, also auch bestimmte (vordefinierte) Kontexte für die angemessene Zuordnung von Intents bzw. für die Verarbeitung und Steuerung von Unterhaltungen berücksichtigen, erscheinen tendenziell intelligenter. Empathischer wiederum stellen sich KIPs dar, die die subjektive Stimmung, Haltung oder Meinung der Nutzer*innen (Sentiments) in ihre Reaktion miteinbeziehen – also nicht nur die Bedeutung des Geäußerten richtig einschätzen, sondern auch die emotionale Einstellung zum jeweiligen Gegenstand des Gesprächs.

Zum Trainieren der Verständnisfähigkeit von KIPs bzw. für die Definition von Intents gibt es unterschiedliche Lösungen – von rein textbasierten Oberflächen, wo Listen von Phrasen für einen Intent eingegeben werden können oder Intents mittels einer eigenen Auszeichnungssprache (wie etwa der Artificial Intelligence Markup Language, AIML) definiert werden, über die Möglichkeit, bestehende Dokumente mit einigermaßen strukturierter Information zu importieren, bis hin zu grafischen Oberflächen in Form von Baumdiagrammen und Ähnlichem. Oftmals werden auch

fertige „Bibliotheken“ mit Standard-Intents (z. B. für Smalltalk oder zentrale Gesprächsdomänen) angeboten, die einfach in das Repertoire eigener KIPs integriert werden können.

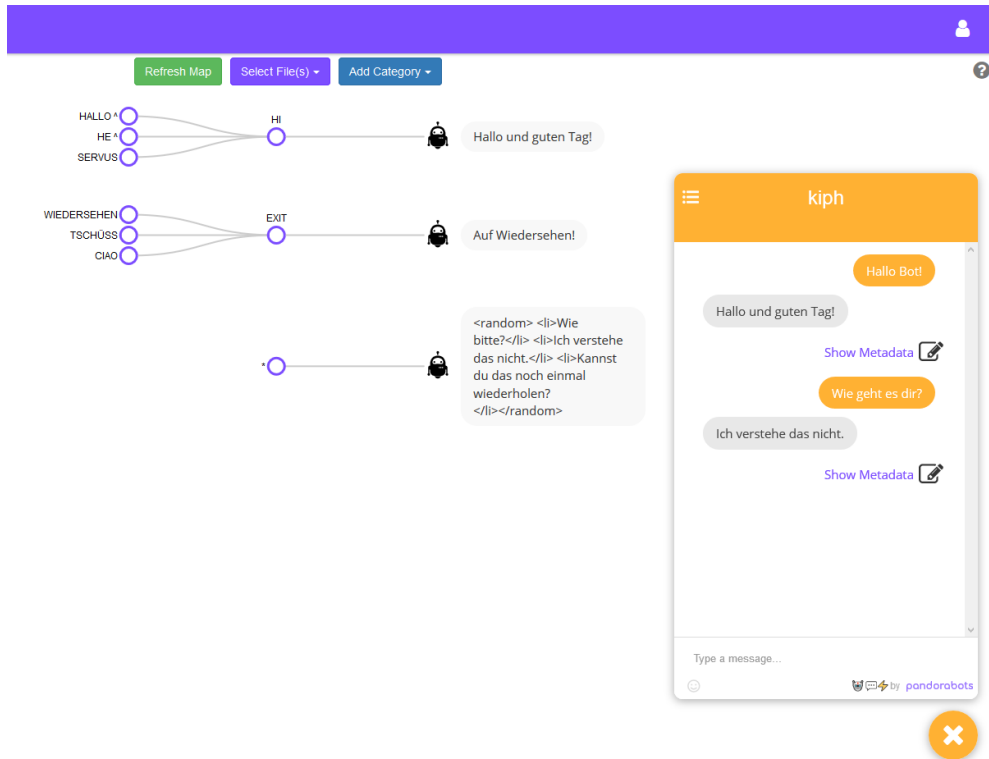


Abb. 1: Ansicht eines (einfachen) „Intents Tree“ in Pandorabots (mit Live-Test; Stand vom 7. Mai 2020)

3.1.3 Gesprächsverläufe

Rein auf bestimmte Intents trainierte KIPs verfügen zwar über eine gewisse Form des Verstehens, sind aber noch nicht in der Lage, flüssige und komplexe Gespräche zu führen. In der Regel wird ein Gespräch mit KIPs auch nicht aus einem bloßen Frage-Antwort-Spiel bestehen, sondern nach bestimmten Strukturen verlaufen. Solche Gesprächsverläufe (meist als *Flows* bezeichnet) werden üblicherweise mit einem visuellen Editor gestaltet; dabei werden einzelne Flows (z. B. Begrüßung mit Kurzeinführung und gegenseitigem „Kennenlernen“) definiert und unterschiedliche Elemente dieses Flows (Gesprächsfragmente, Bedingungen oder „Triggers“, Ereignisse etc.) mittels Drag-and-drop in Form von Ablaufdiagrammen zusammengestellt.

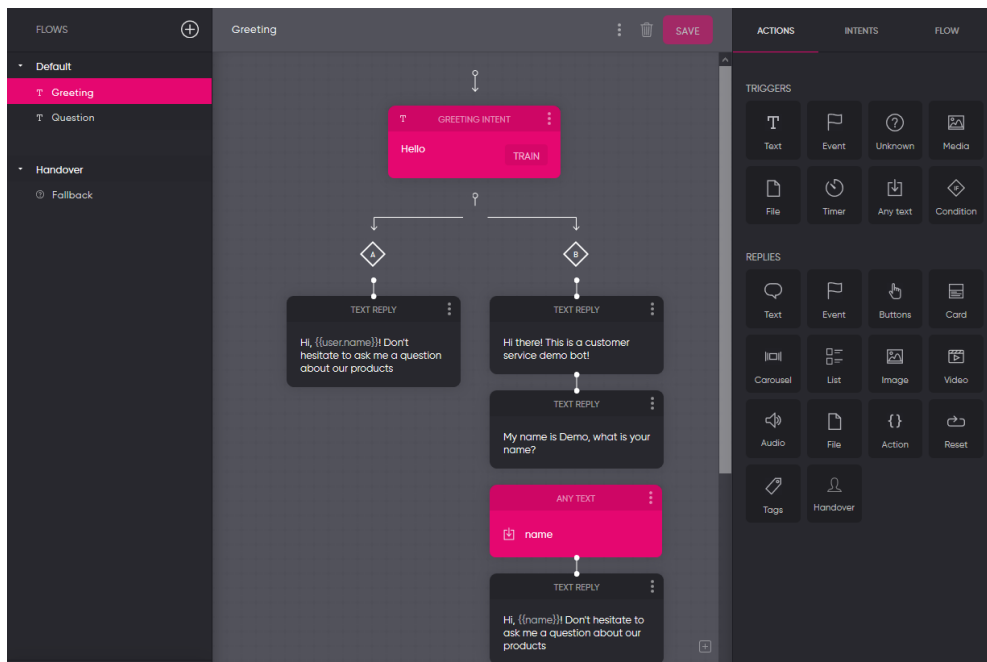


Abb. 2: Der Flow-Editor von flow.ai (Bearbeitung des Begrüßungs-Flows eines Standardtemplates; Stand vom 7. Mai 2020)

Ob die realen Gesprächsverläufe im Sinne des Designs ausfallen oder vom sog. *happy path* abweichen, zeigt dann die Praxis; jedenfalls können und müssen die einzelnen Flows in der Regel nachgebessert und feinjustiert werden. Neben einzelnen Flows wird die Gesprächskunst von KIPs üblicherweise durch einen Fallback-Flow abgerundet, der alle nicht zuordenbaren Nutzer*inneneingaben „abfängt“ und entweder höflich um einen neuen Versuch bittet, Vorschläge für das potentiell Gemeinte macht oder die Konversation an menschliche Gesprächspartner*innen weitergibt („human takeover“).

3.1.4 Interaktionselemente

Bei visuellen bzw. Display-basierten KIPs ist die Interaktion nicht nur auf Reintext beschränkt; vielmehr hat sich eine Reihe von Konversationselementen etabliert, die von den unterschiedlichen Plattformen unterstützt werden und reichhaltigere Dialoge und Interaktionen („rich responses“) erlauben. Abhängig vom jeweiligen Ausgabekanal und der genutzten Entwicklungsplattform zählen dazu etwa:

- Emojis (bunte Ideogramme) oder animierte GIFs und „Stickers“, die Emotionen ausdrücken und Inhalte illustrieren sollen,
- vorgegebene Kurzantworten in Form von Buttons (auch „Quick Responses“, „Quick Replies“ oder „Suggestion Chips“ genannt),
- Bilder, Audio- und Videoinhalte (mit Abspielfunktion),
- Cards (Rahmen mit Bild-Inhalten, Text, Tabellen etc.) bzw. Carousels (Kollektionen)
- (Land-)Karten bzw. Ausschnitte davon,
- ein Persistent Menu (zur „Navigation“) sowie
- eine zeitliche Verzögerung zwischen Eingabe und Antwort(en), um die Interaktion „menschlicher“ wirken zu lassen (optional mit Tipp-Animation oder entsprechenden Symbolen).

3.2 Konkrete Werkzeuge

Das aktuelle – im ersten Halbjahr 2020 vorfindliche – Angebot für die Umsetzung von KIPs ist durch eine Vielzahl von verschiedenen Plattformen und Lösungen gekennzeichnet, aus denen frei gewählt werden kann (und wohl auch muss); die Situation befindet sich irgendwo zwischen anfänglichem Hype mit zahlreichen Start-up-Versuchen und der üblichen Marktkonsolidierung, was die Auswahl einer nachhaltigen Lösung nicht gerade einfach macht.

Im Folgenden wird jedenfalls nur eine Auswahl an Lösungen präsentiert; es wird weder ein Anspruch auf Vollständigkeit erhoben, noch ist davon auszugehen, dass die Ausführungen längerfristige Aktualität beanspruchen können. Aus diesem Grund wurde auch die Entscheidung getroffen, nicht wahllos eine umfassende Liste von am Markt verfügbaren Produkten mit Steckbriefen zu erstellen, sondern einzelne Lösungen anhand von besonderen Merkmalen bzw. Spezialitäten vorzustellen, wobei diese schwerpunktmäßige Illustration gleichzeitig die Auswahl erleichtern und der Orientierung dienen soll. Aufgrund des eher explorativen und nicht auf profunde Praxistests abzielenden Vorgehens im KIPH-Projekt wird hier eher ein Funktionsspektrum aufgezeigt – nicht jedes Feature wird bei jedem Tool erwähnt –, und es werden auch keine expliziten Produktempfehlungen ausgesprochen.

Vorweg ist weiters zu bemerken, dass das Freemium-Lizenzmodell ein weit verbreiteter Standard bei Bot-Services ist: Man kann also Grundfunktionen bzw. ein limitiertes Kontingent an Leistungen gratis nutzen, weiterführende Premium-Funktionen oder der Produktivbetrieb sind allerdings kostenpflichtig.

Mit der im Folgenden vorgenommenen Gliederung sind bereits einige Auswahlkriterien für KIP-Lösungen genannt: sofortige Einsatzbereitschaft, intuitive Erstellbarkeit, On-premises-Verfügbarkeit, kollaborative Erstellung, Marketing-Tauglichkeit, Programmier- und Erweiterbarkeit, spezialisierte Lösungen, Lautsprachbegabung und physische Realisierung. Weitere Kriterien bzw. Gründe für die Wahl eines konkreten Service können sein: Wie soll die Eingabe des „Wissens“ erfolgen (mittels eines visuellen Editors, Code/Markup wie AIML oder eines Flow-

Editors)? Welche Ausgabekanäle werden unterstützt? Was kostet die Nutzung bzw. darf die Nutzung kosten? Gibt es Templates oder bereits vordefinierte Bots für den schnellen Einsatz?

3.2.1 „Fertige“ Bot-Lösungen auf Kommunikationsplattformen

Als Einstieg werden sofort einsetzbare und erweiterbare Bots angeführt, die allerdings auch Nachteile mit sich bringen: Sie sind teilweise kostenpflichtig und bieten im Vergleich zu Entwicklungsplattformen wenig Gestaltungsmöglichkeiten. Im App-Verzeichnis von Slack²² etwa finden sich viele Bots, die einfache Kommunikationsaufgaben übernehmen oder die Organisation und Produktivität von Arbeitsgruppen – insbesondere „remote teams“ – unterstützen. So können etwa mithilfe des „Dailybot“²³ tägliche Umfragen („asynchrone Stand-up-Meetings“) für Teammitglieder durchgeführt werden, um die Motivation zu steigern und Leistungen zu erfassen. Für automatisierte Onboarding-Maßnahmen, die die Aufnahme neuer Mitglieder oder Studierender begleiten sollen, kann etwa der „GreetBot“²⁴ eingesetzt werden. Aber auch Termine, Aufgaben und Auswahl-, Findungs- und Entscheidungsprozesse können in Form von Slack-Bots verwaltet und kommuniziert werden.

Diese Bots werden einfach aus dem App-Verzeichnis in einen sogenannten Slack-Workspace, den man betreibt, installiert, den jeweiligen Gegebenheiten und Erfordernissen entsprechend angepasst und den relevanten Kommunikationskanälen („Channels“) zugewiesen, in welchen die Bots operieren sollen (sie erscheinen dort mit dem Zusatz „APP“, um sie von menschlichen Mitgliedern zu unterscheiden). Meist ist die Nutzung von Slack-Bots in eingeschränktem Umfang kostenlos, für den extensiven Einsatz sind jedoch Nutzungsgebühren zu entrichten.

²² <https://slack.com/apps/>

²³ <https://slack.com/apps/A44PZQW83-dailybot>

²⁴ <https://slack.com/apps/A2LJCT3GR-greetbot>

Für die Demonstration von fertigen Bots in Team-Kommunikationsdiensten wurde Slack aufgrund seiner Popularität herangezogen; ähnliche Lösungen oder „Apps“ bieten jedoch auch andere Anbieter wie beispielsweise Microsoft Teams²⁵ oder Cisco Webex Teams²⁶ in ihren jeweiligen App-Verzeichnissen an.

Auch für einfachere (und eher für den privaten Gebrauch gedachte) Messenger-Dienste wie etwa Facebook Messenger, Kik, Line, Telegram oder Viber sind Bots verfügbar; meist werden diese in eigenen Verzeichnissen (z. B. Botostore²⁷ für Facebook Messenger und Telegram oder bots.kik.com²⁸ für Kik) oder selbst mithilfe von Bots (BotsArchiveBot²⁹ für Telegram) gelistet. Teilweise können solche Bots die Teamkommunikation fördern und die Organisation und Produktivität unterstützen, die überwiegende Mehrheit dient jedoch der Unterhaltung und dem Zeitvertreib. Anstatt auf einen solchen fertigen Bot zurückzugreifen, bietet sich die Nutzung einer Bot-Entwicklungsplattform an, mit der eigene, spezialisierte Bots erstellt werden können und welche die genutzten Messenger als Ausgabekanäle unterstützt.

Für den Hochschulbereich und speziell für die dort eingesetzten Systeme (Web-Redaktionssysteme bzw. Contentmanagementsysteme, Lernmanagementsysteme, Kommunikations- und Kollaborationslösungen bzw. Groupware, Campusmanagementsysteme etc.) wäre es natürlich ebenso wünschenswert, geeignete Bots aus einem Katalog mittels Mausklick in die hauseigenen Systeme integrieren zu können – um z. B. künstliche Tutor*innen oder Lehrassistent*innen als Ansprechpersonen im eigenen Moodle-Kurs oder virtuelle Studienberater*innen auf der Institutshomepage zur Verfügung zu stellen. Aus technologischer Sicht ist eine solche

²⁵ <https://appsource.microsoft.com/de-AT/marketplace/apps?product=teams>

²⁶ <https://apphub.webex.com/teams>

²⁷ <https://botostore.com/>

²⁸ <https://bots.kik.com/>

²⁹ <https://t.me/BotsArchiveBot>

Idee durchaus realisierbar; die meisten Chatbots sind in der Regel als Website-Widget realisiert und können recht einfach mittels HTML-Codebaustein in Webseiten integriert bzw. auch als Moodle-Block umgesetzt werden. Da das Ergebnis jedoch aus mehreren individuellen Bausteinen und Datenquellen zusammengesetzt ist und die verschiedenen hochschulinternen Systeme meist stark an die jeweiligen Anforderungen angepasst sind, muss für die Umsetzung höchstwahrscheinlich in (Eigen-)Entwicklung investiert werden.

3.2.2 Einfache, intuitive Entwicklungsplattformen

Im Gegensatz zu fertig einsetzbaren und auf bestimmten Kanälen verfügbaren Bots legen sich die meisten Bot-Entwicklungsplattformen in der Regel weder auf einen spezifischen Kommunikationskanal noch auf einen bestimmten Funktionsumfang des Bots fest, sondern räumen vielfältige und freie Gestaltungsmöglichkeiten ein. Im Sinne eines Baukastensystems (und eventuell durch Templates unterstützt) können hier beliebige KIPs entworfen und auf unterschiedlichen Kanälen „installiert“ werden. Dabei reicht das Spektrum von recht komplexen Plattformen – komplex sowohl im Sinne der Vielfalt an Funktionen als auch der Bedienbarkeit – bis hin zu sehr einfach gehaltenen Anwendungen, mit denen man recht schnell und ohne intensive Auseinandersetzung vorzeigbare Ergebnisse erzielen kann. Als Beispiele werden im Folgenden drei unterschiedliche Lösungen kurz vorgestellt.

Landbot.io

Die vermutlich intuitivste unter den hier vorgestellten Lösungen zur Entwicklung von KIPs – und auch attraktiv gestaltet – ist Landbot.io, die einen Mindmap-artigen Flow-Editor bietet. Auf einer Leinwand werden Gesprächselemente („blocks“) platziert, der Gesprächsverlauf ergibt sich aus den Verknüpfungen zwischen den Elementen. Landbot.io bietet ausschließlich strukturierte Gespräche; es ist also keine (freie) Chat-Texteingabe vorgesehen und damit auch keine Definition von Intents nötig. Die Interaktion erfolgt vielmehr mithilfe von vordefinierten Antwort-Buttons und verschiedenen Fragetypen, die etwa zur Eingabe des eigenen Namens, der E-Mail-Adresse und anderer Daten, von Zahlen, größeren Texten sowie der

persönlichen Einschätzung mittels Rating- und Bildauswahl-Möglichkeit auffordern.

Der Editor bietet eine Simple- und Advanced-Ansicht, um KIPs bei Bedarf auch mit komplexeren Funktionen auszustatten (z. B. Bedingungen, nutzerspezifische Variablen etc.) oder an andere Web-Dienste anzubinden. Verfügbare Vorlagen und eine Einführungs-“Tour“ mit der Erklärung der basalen Funktionen runden den Eindruck einer zugleich schlichten und praktischen Gestaltungslösung ab.

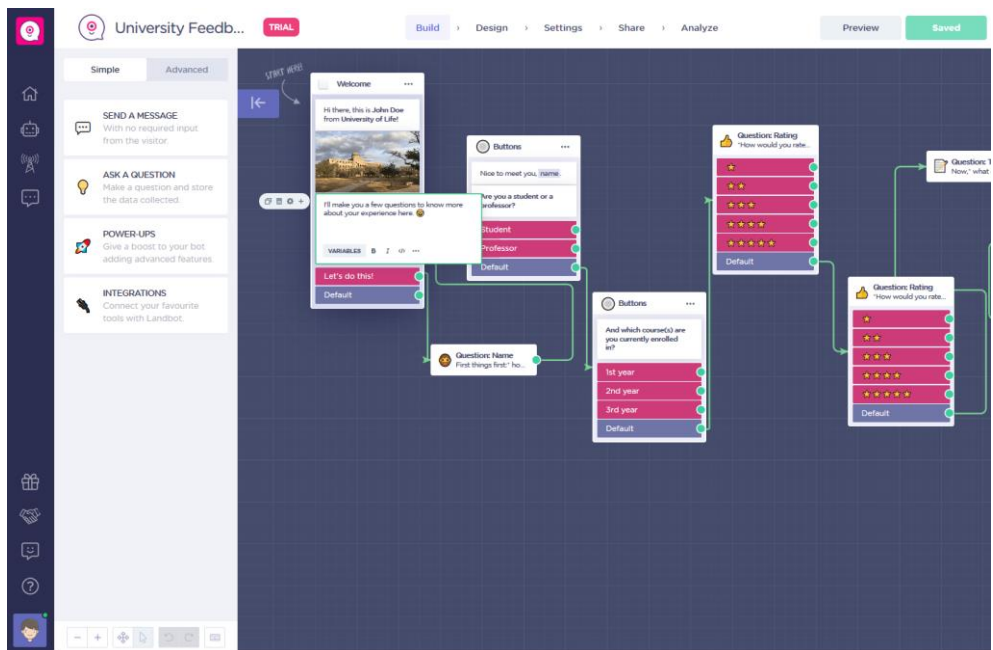


Abb. 3: Bearbeitung der „University Feedback“-Vorlage mit dem Landbot.io-Editor (Stand vom 8. Juni 2020)

Als Kommunikationskanäle werden WhatsApp, Facebook Messenger und andere Kanäle via API (application programming interface) sowie ein optisch gestaltbarer Web-Chat mit verschiedenen Darstellungsmöglichkeiten angeboten. An jeder Stel-

le kann zum Testen eine Web-Vorschau der KIPs aufgerufen werden, und nach der Veröffentlichung stehen (wie bei solchen Diensten üblich) übersichtliche Auswertungen zur Einsicht in die erfolgten Gespräche, Nutzer*innendaten und Chatverläufe bereit.

Flow.ai

Flow.ai wurde bereits oben zur Illustration von Gesprächsverlauf-Editoren herangezogen; ähnlich wie bei Landbot.io steht auch hier für die Gestaltung von Gesprächsverläufen ein (optisch weitaus nüchterner gehaltener, aber ähnlich übersichtlicher) Editor im Leinwand-Stil zur Verfügung. Die wesentlichen Unterschiede zu Landbot.io bestehen darin, dass die Gesprächselemente nicht beliebig platziert und nachträglich mit Pfeilen verknüpft, sondern gemäß der Gesprächslogik nur an bestimmten Stellen „angedockt“ werden können. Weiters stehen nicht nur „statische“ Bausteine (verschiedene „Triggers“ und „Replies“) zur Verfügung; vielmehr können auch „intelligente“ Elemente (Intents) definiert werden, die es erlauben, Texteingaben der Nutzer*innen zu „verstehen“ bzw. zu klassifizieren und in einen Flow zu integrieren. Im Gegensatz zu simplen Lösungen sind hier also Texteingaben möglich, die dann (im Idealfall) bestimmten Intents zugeordnet werden.

Die Definition der Intents erfolgt mittels einfacher Listen von Wörtern und Phrasen in einer eigenen Editor-Ansicht.

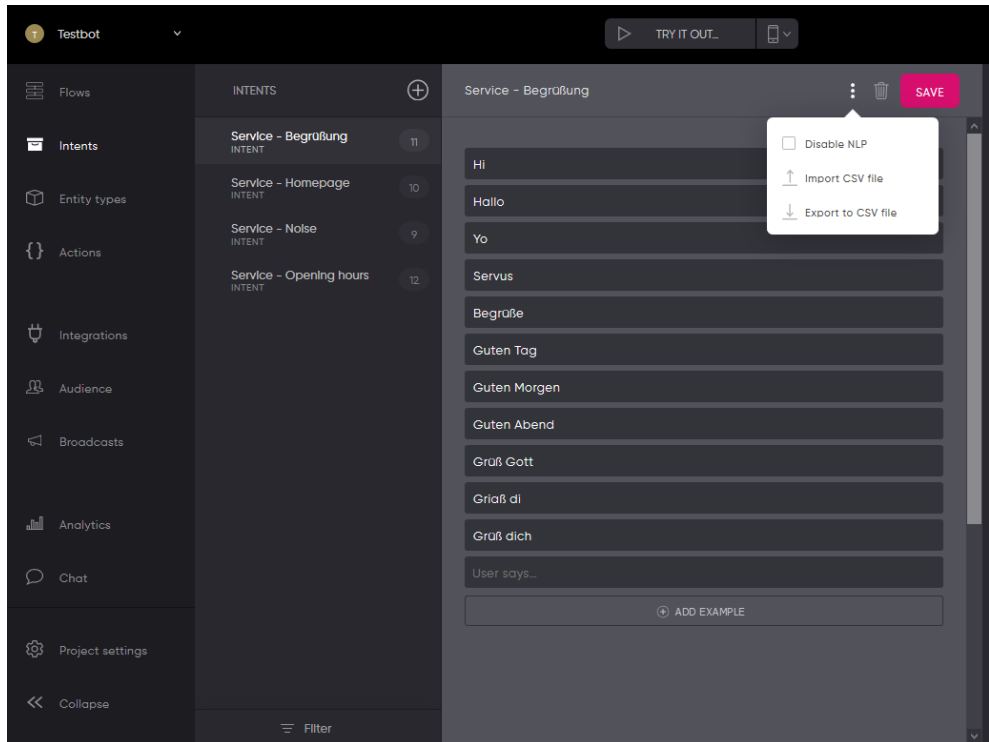


Abb. 4: Intents mit Flow.ai definieren (Stand vom 8. Juni 2020)

Dadurch ist es möglich, Gesprächsverläufe freier zu gestalten, d. h. nicht nach einem generellen Ablaufschema vorzugehen, sondern anhand mehrerer kleiner Flows, die an verschiedenen Stellen des Gesprächs „ablaufen“ können. Spezielle Elemente von Flow.ai (z. B. Entities, Tags oder Actions) erlauben es zusätzlich, KIPs mit einem basalen „Gedächtnis“ auszustatten, Daten aus dem Gespräch zu extrahieren und mit anderen Systemen auszutauschen.

Dialogflow

Dass (auch in der kostenfreien Standardversion verfügbare) umfangreiche Gestaltungsmöglichkeiten nicht im Widerspruch zu einer übersichtlichen Oberfläche stehen müssen, beweist die NLU-Plattform von Google, Dialogflow³⁰. Für die Gestaltung von sogenannten „Agents“ bzw. deren Anbindung an weitere (Google-) Systeme und Services werden hier umfangreiche Möglichkeiten unter einer Oberfläche geboten.

Die Definition von Intents und Entities, deren „matching“ mit Machine Learning (für viele Sprachen) unterstützt wird, ist einfach gehalten, bietet bei Bedarf aber auch feingranulare Funktionen, um die Intent-Erkennung dynamischer und sensibler zu gestalten (Definition von Kontexten, Extraktion von Parametern, Berücksichtigung von bestimmten „Ereignissen“ – im einfachsten Fall etwa das Begrüßungs-Ereignis zu Gesprächsbeginn). Für die Strukturierung von Gesprächen ist kein Flow-Editor vorgesehen; der Fokus von Dialogflow liegt vielmehr auf freien Gesprächen, die mithilfe von angereicherten Intents (d. h. Intents mit Eingabe- und Ausgabekontexten bzw. sogenannten Follow-up-Intents) strukturiert, gesteuert und/oder mit einem „Gedächtnis“ versehen werden.

³⁰ <https://dialogflow.com/>

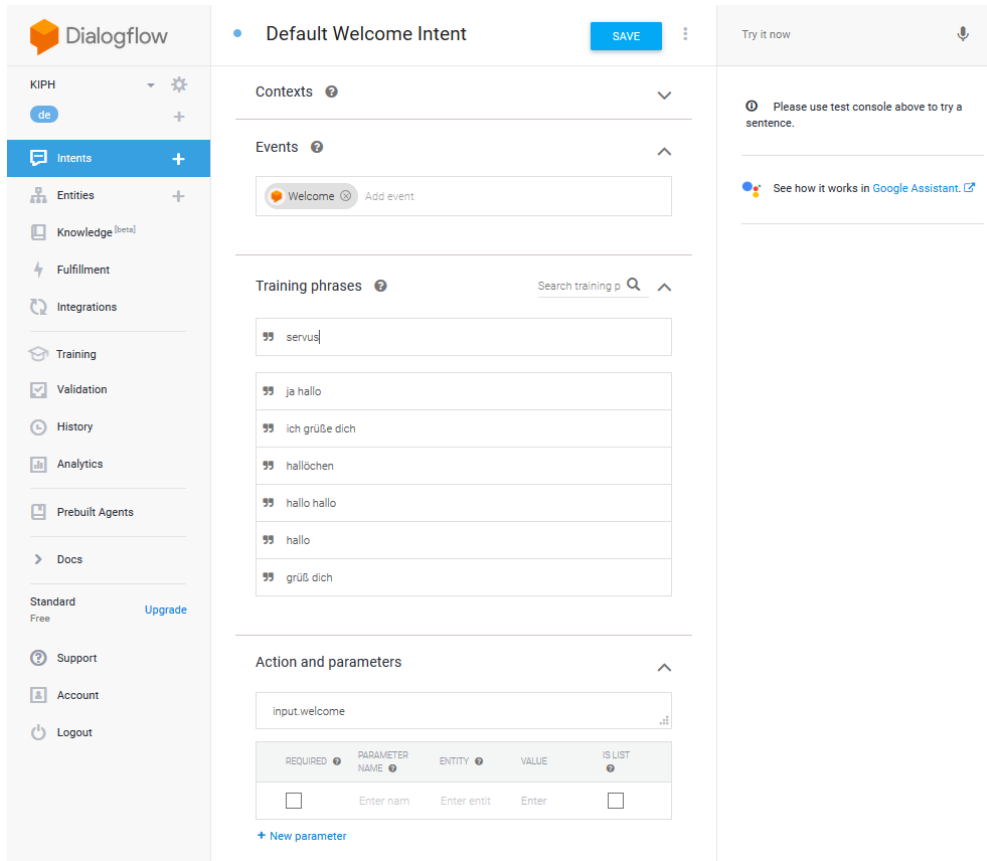


Abb. 5: Gestaltung eines Standard-Begrüßungs-Intents mit Dialogflow (Stand vom 3. Juni 2020)

Um Dialogflow-KIPs mit weiteren Fähigkeiten auszustatten, können Anfragen („fulfillments“) bei externen Webdiensten ausgeführt, Wissensdatenbanken in den Intent-Katalog integriert oder z. B. Wetter- oder Übersetzungsdienste eingebunden werden; zusätzlich können auch Audio-Eingabe und -Ausgabe genutzt werden (s. u.). Für den Einstieg in Dialogflow steht eine umfangreiche Dokumentation mit

Kurzanleitungen und Einführungsvideos zur Verfügung; benötigt wird lediglich ein Google-Account. Die Kontingente (maximale Anzahl der Anfragen bzw. maximale Länge von Audio-Anfragen) der kostenlosen Standard-Version sind recht großzügig bemessen, sodass mit Dialogflow auch ohne finanzielle Ausgaben in der Praxis nutzbare KIPs umgesetzt werden können.

3.2.3 On-Premises-Lösungen

Neben dem Aufwand für die Einrichtung eines Bots und der Komplexität – also sowohl der Bedienbarkeit als auch der „Mächtigkeit“ – der Entwicklungsplattform besteht ein wesentliches Kriterium für die Auswahl der Lösung darin, auf welcher Infrastruktur die KIPs beheimatet sind (bzw. auf welchen Infrastrukturen dessen Komponenten betrieben werden). Im Regelfall sind die KIP-Entwicklungsplattformen als Software-as-a-Service angelegt, d. h. die KIP-Entwicklung und der Betrieb erfolgen im Rahmen eines Cloud-basierten Dienstes. Damit verbunden werden sämtliche Daten vom Betreiber der Plattform verarbeitet; dies betrifft nicht nur die „Inhalte“, also „Wissen“ und Funktionsweise der KIPs, sondern auch die bei der Nutzung anfallenden Daten – Gesprächsverläufe und sämtliche von den Nutzer*innen eingegebenen bzw. (teilweise auch ohne deren Wissen und Kenntnis) übermittelten Daten.

Wenn dies (aus verschiedenen Gründen – z. B. im Hinblick auf den Datenschutz, sensible Informationen oder eingeschränkte Zielgruppen) nicht gewünscht ist, kann man auch auf selbstgehostete Services zurückgreifen. Je nach den individuellen Anforderungen können sowohl die Infrastruktur (Server) an der Hochschule („on premises“) betrieben als auch Software, Datenbanken und sonstige Komponenten von KIPs selbst aufgesetzt und verwaltet werden. Im Gegensatz zu fremdgehosteten bzw. „gemieteten“ Services behält man hier weitestgehend die Kontrolle, muss aber auch – zusätzlich zur inhaltlichen Gestaltung der KIPs – Ressourcen für die Installation und den laufenden Betrieb aufbringen.

Rasa

Eine sehr populäre und umfangreiche Open-Source-Lösung ist Rasa³¹. Neben „Rasa Open Source“ (das eigentliche „machine learning framework for automated text and voice-based conversations“) kann auch „Rasa X“ installiert werden, ein (nicht quelloffener) Werkzeugkasten, mit dessen Hilfe KIPs bereitgestellt, verwaltet, überwacht und verbessert werden können. Rasa Open Source besteht aus einer NLU- und einer Core-Komponente, die eine „dialogue engine for building AI assistants“ darstellt; die NLU-Komponente kann aber auch unabhängig davon genutzt werden.

Rasa Open Source kann on premises installiert (gemäß den Anleitungen für Windows, Mac oder Ubuntu; Python ist jeweils erforderlich) oder auf den größeren Cloud-Anbietern (Amazon S3, Google Cloud Storage, Microsoft Azure Storage) gehostet und betrieben werden. Die Installation des Basissystems und das Aufsetzen und Trainieren von KIPs erfolgt über die Kommandozeile bzw. einen Texteditor, auch Testgespräche mit erstellten KIPs können direkt in der „Rasa Shell“ geführt werden. Demgegenüber ist die Rasa-X-Werkzeugpalette nach der Installation auf einer Weboberfläche erreichbar.

Als Hilfestellungen stehen eine ausführliche Dokumentation, YouTube-Anleitungen und ein Community-Forum zur Verfügung, der Quellcode von Rasa Open Source steht auf Github bereit. Neben Rasa Open Source und Rasa X, die gratis verfügbar sind, kann auch das kostenpflichtige „Rasa Enterprise“-Service in Anspruch genommen werden, das u. a. Analytics-Funktionen, Zugriffskontrolle, Single Sign-On und professionellen Support bietet.

³¹ <https://rasa.com>

Botpress

Ebenfalls für Windows, Mac und Linux verfügbar und weitgehend mit Rasa vergleichbar ist Botpress, eine weitere Open-Source-Lösung zur Installation auf der eigenen Infrastruktur. Wie schon im Namen angedeutet, erhebt Botpress den Anspruch, das „Wordpress“ für Bots zu sein, also eine ähnlich brauchbare Lösung für Bots wie Wordpress für Weblogs zu sein. Seit Anfang 2020 präsentiert sich Botpress in einem kommerzielleren Erscheinungsbild³²; neben der kostenfreien Open-Source-Version wird für größere Unternehmen eine kostenpflichtige Enterprise-Version mit zusätzlichen Funktionen angeboten.

Die im vorliegenden Kontext interessantere Open-Source-Version enthält verschiedene Basis-Module, wie sie auch von anderen Lösungen bekannt sind (u. a. ein NLU-Modul für das „Verstehen“ von Intents und Entities bzw. „Slots“, ein Q&A-Modul für einfache Fragen-Antwort-Paare, einen Emulator für Testgespräche, einen Flow-Editor für komplexe Gesprächsverläufe, einen Code-Editor für Anbindungen und eine basale Analytics-Übersicht). Die Oberfläche zum Bearbeiten von Bots nennt sich Studio.

³² <https://botpress.com>; vormals: <https://botpress.io>

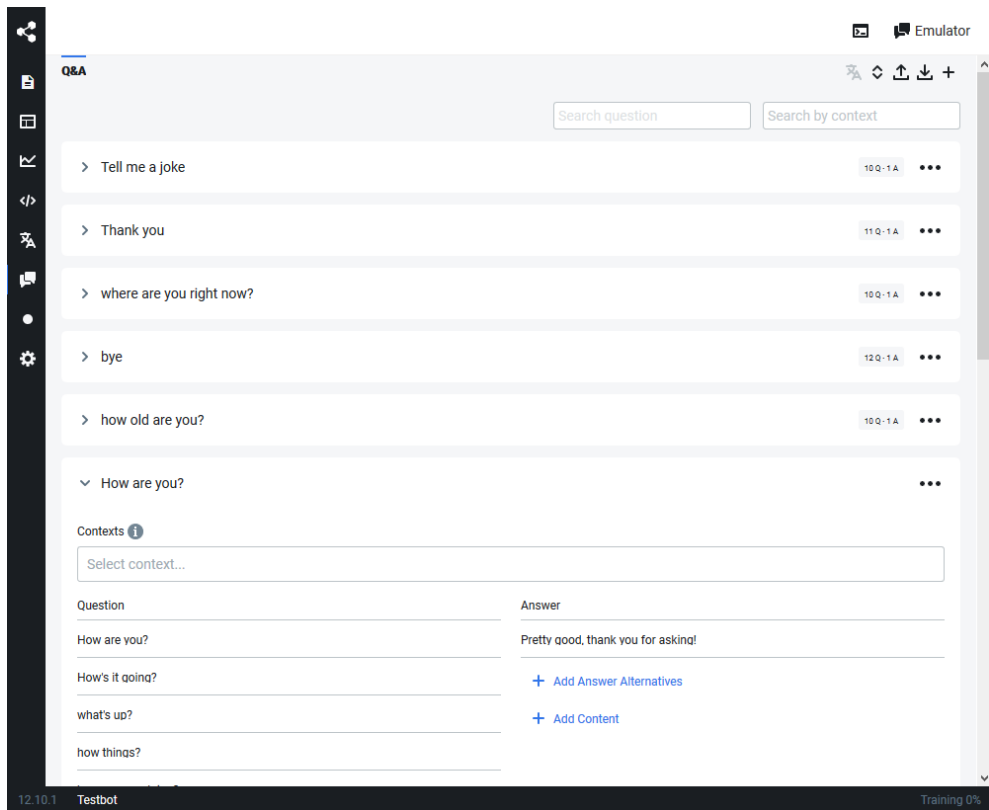


Abb. 6: Entwicklungsoberfläche „Studio“ von Botpress (Version 12.10.1 unter Windows); hier: Bearbeitung von Questions & Answers

Botpress wird als ZIP-Archiv zum Download angeboten und über eine ausführbare Programmdatei gestartet; am Rechner (localhost) ist Botpress dann über den Webbrowser erreichbar, kann lokal genutzt und später auch „deployed“ werden, um die eigenen Botpress-KIPs der Öffentlichkeit zur Verfügung zu stellen.

Auf den Webseiten von Botpress stehen eine umfangreiche Dokumentation und Tutorials zur Verfügung, weiters werden Codebeispiele und ein SDK (Software

Development Kit, also eine Programmier-Werkzeugkiste) zur Entwicklung eigener Module bereitgestellt.

Ana

Etwas weniger komplex und mit geringerem Umfang an Möglichkeiten und Funktionen präsentiert sich Ana³³, nach eigenen Angaben das „World’s First Open-Source ChatBot Framework“. Ein wesentlicher Vorteil von Ana ist die unkomplizierte Installation (als Desktop-App für verschiedene Betriebssysteme, die eine „Studio“-Oberfläche zur Erstellung von KIPs inklusive Chat-Simulator sowie Konfigurationsmöglichkeiten und Analytics eigener „Ana Chat Server“ bietet); die Erstellung von KIPs (die sich auf einen Flow-Editor beschränkt, also kein „Verständnis“ von Texteingaben beinhaltet) gestaltet sich ebenfalls sehr einfach.

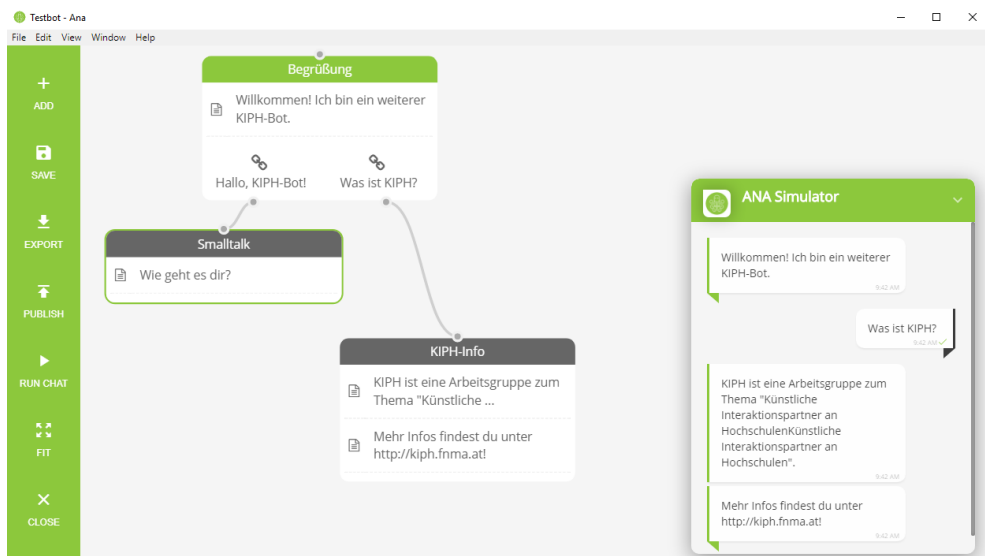


Abb. 7: Das Ana-Studio mit Flow-Beispiel unter Windows (Version 0.1.8)

³³ <https://www.ana.chat/>

Die jeweilige (spärliche) Dokumentation findet sich auf den Github-Repositorien des indischen Entwicklungsunternehmens für die einzelnen Komponenten³⁴. Als öffentlicher Kommunikationskanal wird von Haus aus nur ein mittels des Ana Chat Server realisierter Web-Chat unterstützt, weiters können Ana-Bots mithilfe des SDK in eigene iOS- oder Android-Apps integriert werden.

Weitere Möglichkeiten

Neben den erwähnten Komplettlösungen für den Betrieb bzw. die Entwicklung auf der eigenen Infrastruktur sind im Web auch viele Lösungsansätze, Codebausteine und Programmbibliotheken verfügbar, mit denen man Bots zusammenbasteln kann, ohne die Cloud-Services bestimmter Anbieter*innen in Anspruch nehmen zu müssen; neben Programmierkenntnissen ist hierzu aber auch ein Grundverständnis des Betriebs von webbasierten Anwendungen hilfreich.

Darüber hinaus sind auch Hybridlösungen möglich, wenn etwa bestimmte Cloud-Komponenten (z. B. Machine-Learning-unterstütztes Natural Language Understanding oder Sprachein- und/oder -ausgabe) für eine*n ansonsten „on premises“ betriebene*n KIP genutzt werden.

3.2.4 Kollaboration und Austausch

Eine zentrale Anforderung besteht oftmals darin, dass mehrere Personen (etwa Kolleg*innen aus der Abteilung, dem Institut oder der Hochschule, aber auch in interinstitutionellen Teams) an der Entwicklung von KIPs zusammenarbeiten oder diverse Entwicklungen teilen und weitergeben möchten. Viele Bot-Entwicklungsplattformen bieten zwar die gemeinsame Bearbeitung und Kollaboration an – meist aber nur im Rahmen von kostenpflichtigen Premium-Mitgliedschaften, die den Zugriff auf Ressourcen für mehrere Einzelnutzer*innen bzw. Teams ermöglichen.

³⁴ <https://github.com/nowfloats>

SAP Conversational AI

Ein gegenteiliges Beispiel ist SAP Conversational AI³⁵, das sehr Community-orientiert aufgebaut ist. Bots können hier einfach mit einzelnen Nutzer*innen geteilt und anschließend gemeinsam bearbeitet werden (Voraussetzung ist ein Account auf der CAI-Plattform). Es ist weiters möglich, verschiedene Organisationen zu erstellen (wahlweise öffentlich zugängliche oder private, d. h. nur für Mitglieder einsehbare), denen man angehört und die man wiederum in kleinere Teams unterteilen kann. KIPs (die ebenfalls individuell als öffentlich oder privat erstellt werden können) können dann diesen Organisationen oder einzelnen Teams zugewiesen werden, wodurch alle Mitglieder – gemäß der ihnen zugeteilten Rechte – Zugriff haben.

Darüber hinaus ist es auch möglich, KIPs, auf die man Zugriff hat (eigene, geteilte oder öffentliche), und ebenso Teile davon zu „forken“. Begriffsgeschichtlich in der Softwareentwicklung beheimatet, sind „Forks“ gewissermaßen eigene Kopien oder Abspaltungen, die unabhängig von ihrem Ursprung weiterentwickelt werden; dies betrifft bei SAP Conversational AI sowohl ganze Bots, aber auch einzelne Skills (ganze Konversationskomponenten), Intents und Entities. Jedes dieser Elemente kann in eigene Projekte kopiert, dort weiterverwendet oder angepasst werden – für Intents steht auf der Bearbeitungsoberfläche sogar eine Suchfunktion zur Verfügung.

Diese Möglichkeit bietet insgesamt ein weitaus flexibleres Instrument als die Verfügbarkeit (starrer) Templates.

³⁵ <https://cai.tools.sap/>

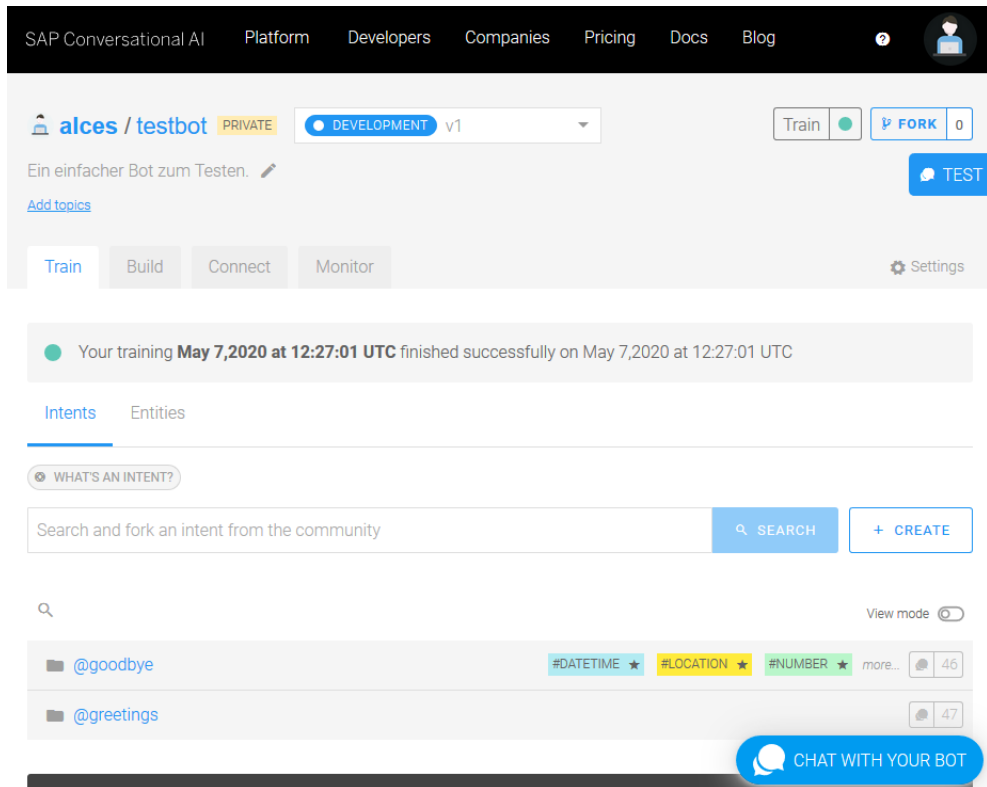


Abb. 8: Intent-Übersichtsseite in SAP Conversational AI mit der Such- und „Fork“-Möglichkeit freigegebener Intents (Stand vom 7. Mai 2020)

Pandorabots

Eine einfache, aber elegante Lösung bietet Pandorabots durch ein besonderes Format (AIML). Pandorabots³⁶ ist eine bereits ältere (und ausgereifte) Lösung mit

³⁶ <https://home.pandorabots.com/>

einem speziellen, textbasierten Ansatz; sie präsentiert sich als die „leading conversational AI platform“ und zeichnet auch für den relativ bekannten und gesprächstauglichen Mitsuku-Chatbot³⁷ verantwortlich. Neben der Möglichkeit, einen visuellen Editor („Intents Tree“ – vgl. die obige Abbildung 1 – und „Chat Design“) zu benutzen, können Pandorabots auch und vor allem mittels AIML (Artificial Intelligence Markup Language) befüllt bzw. bearbeitet werden, also in Form einfacher Textdateien, die die Wissensbestände und Anweisungen in einem (zwar gewöhnungsbedürftigen, aber nicht sonderlich komplexen) XML-Format beinhalten. Der wesentliche Vorteil dieses Textformats ist seine einfache Teil- und Weiterverwendbarkeit: Die Dateien können nicht nur als ZIP-Datei heruntergeladen, lokal bearbeitet, mit anderen geteilt und selektiv in eigene Projekte hochgeladen werden, sondern auch direkt aus der Plattform in ein Github-Repository übertragen werden, wo die systematische Zusammenarbeit an AIML-Dateien (und folglich an KIPs) im Stil von Github stattfindet. In diesem Sinn werden auch kostenlose AIML-Bibliotheken als Vorlagen zum Weiterverwenden öffentlich auf Github bereitgestellt (z. B. „Rosie“ als Chatbot-Basis³⁸).

³⁷ <https://www.pandorabots.com/mitsuku/>

³⁸ <https://github.com/pandorabots/rosie>

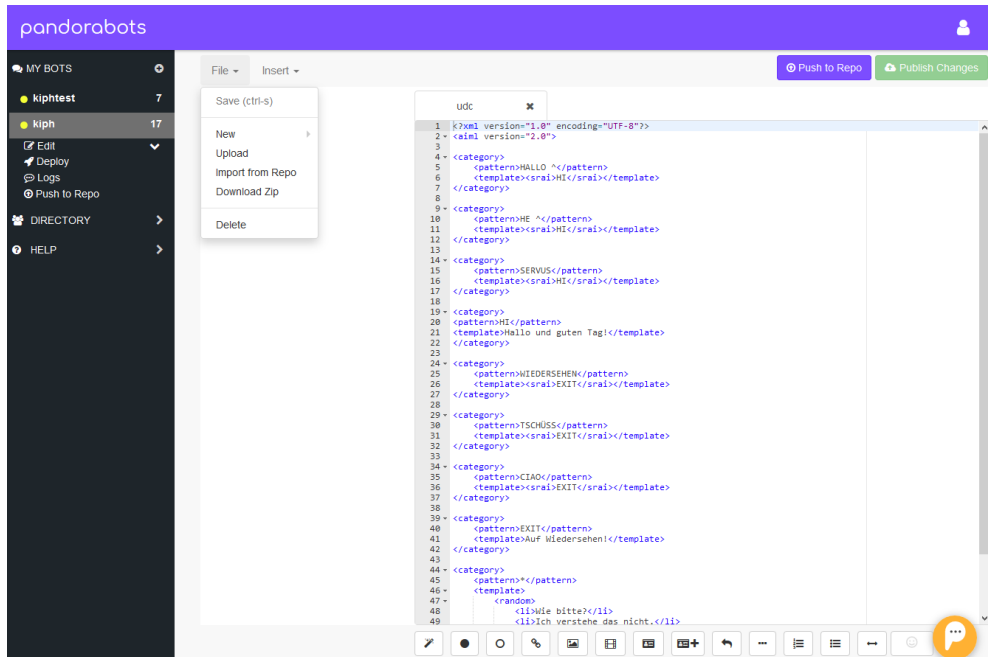


Abb. 9: Code-Editor von Pandorabots mit verschiedenen Import-/Exportfunktionen (Stand vom 7. Mai 2020)

Dieser Ansatz bietet somit nicht nur eine einfache Möglichkeit der Kollaboration, sondern auch – sofern ein solches gewünscht ist – ein hohes Maß an Transparenz; die komplette inhaltliche Grundlage („Wissen“ und „Persönlichkeit“) von KIPs steht auf Wunsch als „Open Source“ zur Verfügung. Pandorabots bietet damit eine Alternative zur bloßen Backup-Funktion, die zumeist lediglich den Export für die Wiederverwendung in der gleichen Lösung umfasst.

3.2.5 Marketing und Lead Generation mit KIPs

Ganz andere Prioritäten kommen bei marketing-tauglichen KIPs zum Zug. So ist etwa der Facebook Messenger³⁹ aktuell einer der populärsten Messenger-Dienste und zugleich auch eine beliebte Bot-Plattform. Da die Nutzung des Facebook Messenger nur mit einem Facebook-Account möglich ist und dieser Account wiederum die Eingabe einer Reihe persönlicher Daten bedingt, die für die Betreiber von Bots auf dieser Plattform einsehbar sind, gewinnt man auf diesem Weg mehr (Kontakt-)Informationen über die menschlichen Gesprächspartner*innen, als dies üblicherweise möglich ist. Dieser Umstand wird zumeist von kommerziellen Unternehmen genutzt, um Interessent*innen bzw. potenzielle Kund*innen zu erfassen („Leadgenerierung“) und mit ihnen in Kontakt zu bleiben (automatisiertes Marketing). Genau hierfür gibt es eine Reihe von maßgeschneiderten Lösungen, die sich primär auf die Möglichkeiten des Facebook Messenger stützen; drei sehr populäre Dienste sind etwa Chatfuel⁴⁰, ManyChat⁴¹ und MobileMonkey⁴².

Zur Nutzung dieser Dienste benötigt man einen Facebook-Account (der für das Login verwendet wird); weiters müssen eine Facebook-Seite eingerichtet und dem Dienst einige Berechtigungen dafür erteilt werden (Werbeanzeigen, Seitenverwaltung, Unterhaltungen). Die drei genannten Dienste sind durchwegs kommerziell orientiert und nicht experimentell angelegt; gemäß dem Freemium-Preismodell können die Basisfunktionen kostenlos genutzt werden, spezielle Funktionen sind allerdings kostenpflichtigen Abonnements vorbehalten.

Das Anlegen von Chatbots ist bei den genannten Lösungen durchwegs einfach gestaltet – der Schwerpunkt liegt nicht auf freier und intelligenter Konversation, sondern vielmehr auf der Übermittlung von Messages oder Auskünften mit meist

³⁹ <https://www.messenger.com>

⁴⁰ <https://chatfuel.com/>

⁴¹ <https://manychat.com/>

⁴² <https://mobilemonkey.com/>

vordefinierten Interaktionsmöglichkeiten. Der „Bot Builder“ von MobileMonkey etwa bietet die Möglichkeit, Chatbots im Dialog- oder im Q&A-Format zu gestalten, während Chatfuel-Bots aus Blöcken aufgebaut werden und mit einfachen Regeln (Nachrichten oder Blöcke als Antwort auf bestimmte Phrasen) trainiert werden; ManyChat wiederum bietet für die Gestaltung von Gesprächsabläufen wahlweise einen einfachen „Basic Builder“ oder einen grafischen „Flow Builder“.

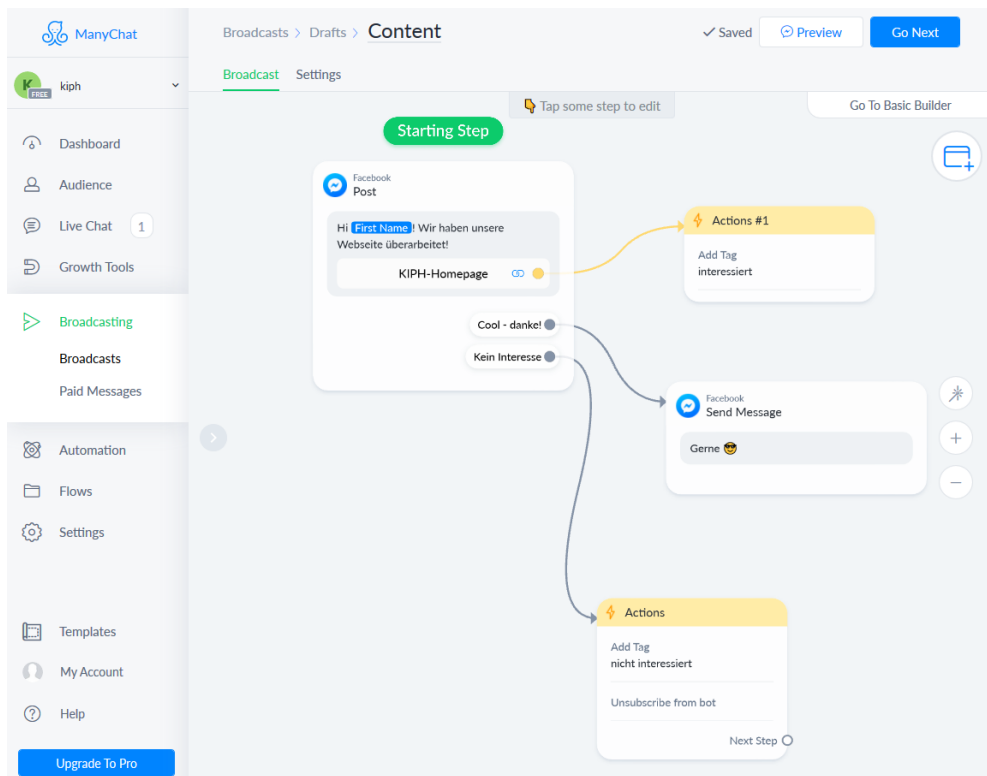


Abb. 10: ManyChat: Gestaltung eines „Broadcasts“ (Marketing-Instrument) mit dem Flow Builder (Stand vom 12. Mai 2020)

Haben sich Nutzer*innen auf eine Interaktion mit solchen Facebook-Messenger-KIPs eingelassen, werden einige basale Daten des Facebook-Accounts übernommen (Name, Geschlecht, Sprache, Zeitzone, Facebook-Profilseite etc.) sowie einige spezifische Daten der Konversation protokolliert (Zeitpunkt und Gesprächsverlauf); zusätzlich können die KIP-Betreiber*innen in der „Kundenkartei“ auch beliebige Notizen oder frei definierbare Tags und Attribute hinzufügen. Die Interaktion mit den KIPs erfolgt üblicherweise mit dem Facebook Messenger (Weboberfläche oder App), zusätzlich kann auch ein Messenger-Widget oder Pop-up für eigene Webseiten generiert oder eine „Landing Page“ (einfache Webseite mit Chat-Bereich) eingerichtet werden. Als einzige Ausnahme der drei Lösungen bietet MobileMonkey ein gewissermaßen „neutrales“ Webchat-Widget („OmniChat“), das keine Facebook-Authentifizierung erfordert; die benötigten Kontaktdaten für die weitere Interaktion können bei Bedarf aber mittels Formular abgefragt werden.

Die solcherart gewonnenen „Leads“ können in Gruppen („Audiences“ oder „Segments“) unterteilt werden, um eine zielgruppenspezifische Anschlusskommunikation zu führen – entweder via Facebook Messenger, aber teilweise auch mittels SMS und E-Mail.

Für automatisiertes Marketing gibt es unterschiedliche Verfahrensweisen (vordefinierte Abfolgen, getriggerte bzw. zeit- oder durch externe Systeme gesteuerte Nachrichten etc.), die unterschiedliche Namen tragen – etwa „Broadcasts“ (ManyChat), „Campaigns“ (Chatfuel) oder „Drip Campaigns“ sowie den „Chat Blaster“ und „RSS Blaster“ (MobileMonkey).

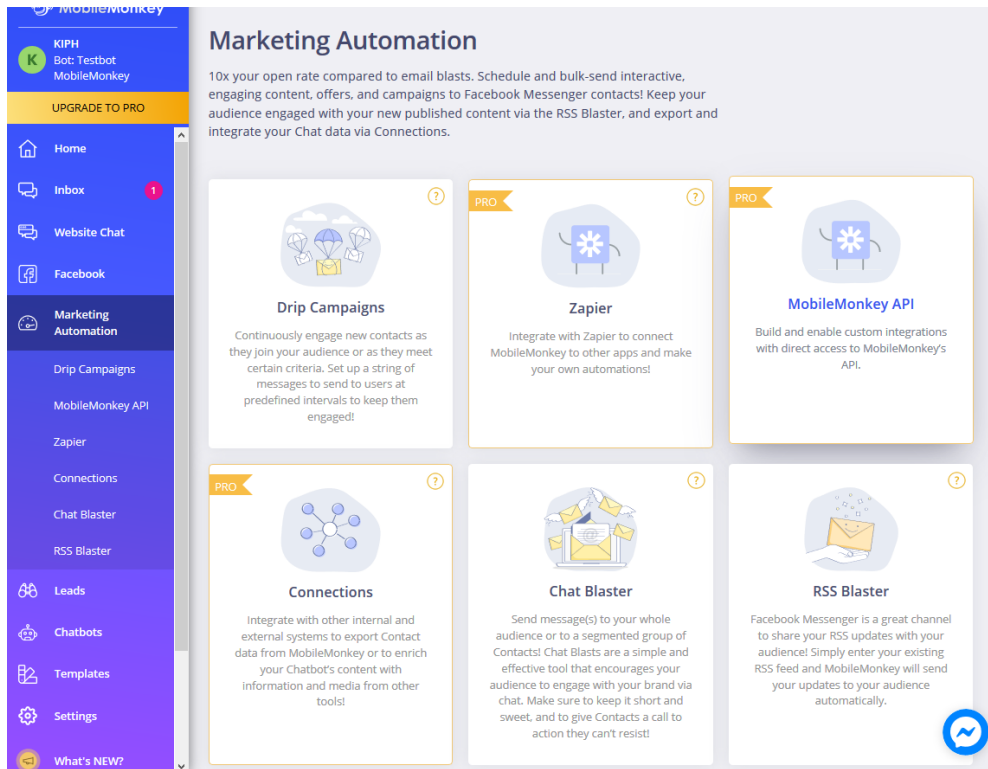


Abb. 11: Marketing-Werkzeuge bei MobileMonkey (Stand vom 12. Mai 2020)

Von Bedenken in Bezug auf den Datenschutz abgesehen, eignen sich die genannten Lösungen augenscheinlich weniger für den Einsatz im Lehrbetrieb, wo man besser beraten ist, mit der eigenen „Audience“ auf anderen Kanälen zu interagieren. Sie scheinen jedoch für die Betreuung von Studieninteressierten oder Alumni relevant – sofern die Zielgruppe Facebook (bzw. Facebook Messenger) nutzt.

3.2.6 Bots an Plattformen anbinden und programmieren

Auf dem Facebook Messenger sind Bots generell sehr willkommen; die bereits 2016 veröffentlichte „Messenger Platform“⁴³ lädt ein, (selbst entwickelte) KIPs mithilfe verschiedener technischer Komponenten in den Messenger zu bringen, woran die zuvor genannten Dienste – mit ihren nutzer*innenfreundlichen Oberflächen zur inhaltlichen Gestaltung von KIPs – andocken. Im Gegensatz dazu ist beispielsweise (das ebenfalls zu Facebook gehörende) WhatsApp⁴⁴ mit Bezug auf Bots und automatisierte Kommunikation weitaus zurückhaltender: Dort setzt man auf (teils kommerzielle) „Business Solutions“. Die meisten Messenger- und Kommunikations-Dienste bieten jedoch – ebenso wie der Facebook Messenger – Möglichkeiten der technischen Anbindung an, etwa Programmierschnittstellen (APIs) oder sogenannte Webhooks, um Kommunikation auch „von außerhalb“ des Dienstes führen, verarbeiten und steuern zu können. „Hinter“ einem Messenger-Account oder Profil muss also nicht immer eine Person sitzen, die Gespräche im jeweiligen Client⁴⁵ initiiert oder beantwortet; auch Computerprogramme auf Servern können diese Aufgabe übernehmen und nach bestimmten Regeln oder auf Ereignisse kommunikativ (re-)agieren und Inhalte in den Dienst „einspeisen“. (Bot-Entwicklungsplattformen leisten im Grunde nichts anderes, als mithilfe von einfachen Oberflächen solche Computerprogramme zu erstellen und mit den Bot-Plattformen zu verknüpfen.)

Auf diese Weise lassen sich nicht nur automatisierte Kommunikationslösungen und programmierte Bots realisieren, sondern auch externe Dienste und Systeme einbinden – etwa, um Informationen in anderen Systemen abzurufen (z. B. Sprechstundenzeiten oder Prüfungsergebnisse aus einem Campusmanagementsystem), dort zu

⁴³ <https://developers.facebook.com/docs/messenger-platform/>

⁴⁴ <https://www.whatsapp.com/>

⁴⁵ „Client“ bezeichnet das Programm, mit dem der Kommunikationsdienst genutzt wird – also z. B. die Facebook-Messenger-App am Smartphone oder auch der Webbrowser am Desktop-Rechner für einen Web-Chat.

speichern (z. B. Lernfortschritte dokumentieren) oder andere Services zu integrieren (z. B. einen Ausschnitt des Campusplans einblenden).

Grundsätzlich sind hierfür Programmierkenntnisse und entsprechende Ressourcen (Server bzw. entsprechende Services) erforderlich; im Web bzw. auf diversen Coding-Plattformen sind allerdings Frameworks und Bibliotheken für unterschiedliche Programmiersprachen zu finden, die zumindest basale Funktionen (Anbindung an den Kommunikationsdienst und Standard-Messaging-Funktionen) bieten und eine relativ einfache Entwicklung ermöglichen.

Ganz basale Beispiele wären etwa Twitterbots, die zunächst mit einem eigenen Twitter-Account verknüpft werden müssen und nach bestimmten Regeln und Vorgaben Inhalte tweeten oder retweeten; solche Twitterbots können in gängigen Programmiersprachen wie PHP oder Python geschrieben sein, aber auch einfacher (z. B. mittels Google-Tabelle⁴⁶) realisiert werden.

Ein kompletter Bot, der mittels „Adapters“ an unterschiedliche Kommunikationsplattformen angebunden, aber auch über die Kommandozeile angesprochen werden kann, ist Hubot⁴⁷. Ursprünglich als Assistent für das Github-Entwicklungsteam entworfen, steht Hubot nun zur freien Weiterverwendung zur Verfügung und kann mit Funktionen aus der Community erweitert und angepasst werden.

Eine „ansprechende“ Form für die Erstellung von Bots bietet etwa der Messenger Telegram⁴⁸: Anstatt (wie üblich) anhand der Dokumentation und der Account-Einstellungen vorzugehen, „spricht“ man einfach mit einem speziellen Bot (dem „Botfather“) und nutzt die vorgegebenen Befehle zur Erstellung und Verwaltung von Bots; für die eigentliche Bot-Entwicklung muss dann die Dokumentation der

⁴⁶ Siehe hierzu die Ressourcen und Anleitung eines gewitzten amerikanischen Hochschullehrenden: <http://www.zachwhalen.net/posts/how-to-make-a-twitter-bot-with-google-spreadsheets-version-04/>

⁴⁷ <https://hubot.github.com/>

⁴⁸ <https://telegram.org/>

Bot-API⁴⁹ konsultiert oder ein Telegram-Bot-Erstellungsdienst wie etwa Manybot.io⁵⁰ in Anspruch genommen werden. Telegram-Bots sind üblicherweise weniger als Partner*innen für freie Konversationen konzipiert, sondern übernehmen spezielle Aufgaben, etwa die Übermittlung spezifischer Informationen und Aktionen, aber auch die Durchführung interaktiver Übungen (Umfragen, Quiz oder HTML5-basierte Spiele).

Zu guter Letzt können KIPs selbstverständlich auch bei darauf spezialisierten Agenturen („Chatbot Agencies“ oder Web-Agenturen) in Auftrag gegeben werden, die eine individuelle Entwicklungsoberfläche bereitstellen und/oder das KIP-Design entwickeln. Dass diese Möglichkeit hier nur beiläufig erwähnt wird, soll nicht darüber hinwegtäuschen, dass sie häufig in Anspruch genommen wird; sie ist jedenfalls nicht die kostengünstigste.

3.2.7 Spezialisierte Lösungen

Umfrage-Bot „Hubert“

Eine spezielle Anwendung von KIP verkörpert „Hubert“⁵¹, der die Durchführung von Umfragen in Form eines strukturierten Chatverlaufs mit einem Bot ermöglicht. Anstatt (wie großteils üblich) unpersönliche und wenig ansprechende Webformulare auszufüllen, werden die Nutzer*innen von „Hubert“ zu ihrer Einschätzung befragt. Hubert ist derzeit auf Kund*inneneinschätzungen, Mitarbeiter*innen-Umfragen (HR) und Umfragen in Bildungssettings spezialisiert, wobei in letzteren Feedback wahlweise nach drei Vorlagen („Stop/Start/Continue“-Methode, „Net Promoter Score“, also Grad an subjektiv empfundener Weiterempfehlbarkeit, sowie der wertschätzenden Methode „2 stars and a wish“) eingeholt werden kann.

⁴⁹ <https://core.telegram.org/bots/api>

⁵⁰ <https://manybot.io/>

⁵¹ <https://hubert.ai>

4. Fill in the gaps

Almost done!
Hubert needs a few inputs to complete the conversations he will have with the respondents. Fill in the gaps and see how your changes affect the chat in the preview.

ADDITIONAL INVITATION MESSAGE (OPTIONAL)

E.g. 'This is our new way of collecting feedback!'

HOW SHOULD HUBERT REFER TO YOU? ?

*Michael

WHAT ARE YOU EVALUATING? ?

*the course

Preview chat

Hubert

What is the first thing you come to think of that is working well with _____?

Respondent

The layout of our assignments is spot on

Hubert

What is the second thing you come to think of that is working well with it?

Respondent

Hmm. Also that we get to schedule our week as we want

Back **Next**




Abb. 12: Umfrage-Bot-Erstellung mit Hubert in fünf Schritten (hier Schritt 4: abschließende Eingabe von Umfrageparametern und Vorschau; Stand vom 28. Mai 2020)

Es ist anzumerken, dass Hubert in der (kostenpflichtigen, aber gratis testbaren) Basisversion nur wenig Gestaltungsspielraum erlaubt – hier können keine individuellen Fragen eingegeben werden -, und dass ein solcher Umfragebot auch mit anderen Lösungen, die einen Gesprächsverlauf zu designen erlauben, nachgebaut werden kann. Auch versteht Hubert aktuell (noch) nicht Deutsch. Hubert ist allerdings sehr einfach zu bedienen, ansprechend gestaltet und bietet im Gegensatz zu Bot-Baukästen praktische Zusatzfeatures (etwa E-Mail-Einladung und QR-Code zur Umfrage in Form einer Chat-Webseite) sowie eine Auswertung der Ergebnisse, die je nach Preismodell unterschiedlich ausfällt.

QnA Maker

Beispiele für den umgekehrten Ansatz (also wenig strukturierte bzw. im Verlauf vorgegebene, dafür jedoch mehr auf punktuelle Informationsgewinnung abzielende Gespräche) sind einfache Frage-Antwort-KIPs, die auf Fragen mit bestimmten Informationen antworten.

„From data to bot in minutes“ – und das ganz ohne Programmierkenntnisse – verspricht etwa der QnA Maker von Microsoft⁵². Obwohl er Teil der (mitunter recht komplex anmutenden) Microsoft Azure Services ist, bietet der QnA Maker eine schlichte, aufgeräumte Oberfläche, auf der man einen Frage-Antwort-Bot erstellen und mit Wissensbeständen („Knowledge bases“) versehen kann.

⁵² <https://www.qnamaker.ai/>

QnA Maker My knowledge bases Create a knowledge base

Tes... EDIT PUBLISH SETTINGS Save and train Test

Knowledge base

Search the knowledg 93 + [grid icon]

Context	Question	Answer
^ Editorial		
Was ist KIPH? L	Homepage X	https://kiph.fnma.at [trash icon]
Homepage	Wo ist die Homepage? X	
	Webseite X	
	URL X	
	+ Add alternative phrasing	
Was ist KIPH? L	Was ist KIPH? X	KIPH (Künstliche Interaktionspartnern an Hochschulen) ist eine Arbeitsgruppe, die sich mit dem Einsatz von Bots an Hochschulen auseinandersetzt. [trash icon]
Homepage	+ Add alternative phrasing	Homepage

Abb. 13: Oberfläche zur Befüllung einer „Knowledge base“ im QnA Maker (Stand vom 22. Januar 2020)

Hierbei können Fragen bzw. Fragevarianten und darauf zutreffende Antworten entweder mit dem Webeditor eingegeben und bearbeitet oder aus bestehenden Res-

sourcen (CSV- oder TSV-Dateien, Excel-Tabellen, FAQ-Listen etc.) via Upload oder URL importiert werden; optional können Frage-Antwort-KIPs auch mit den hauseigenen Small-Talk-Beständen („Chit chat“) in verschiedenen Sprachen angereichert werden, um sie etwas persönlicher und gesprächiger zu gestalten. Die Fragen der Nutzer*innen müssen nicht exakt mit den vorgegebenen Fragestellungen übereinstimmen; vielmehr werden die KIPs mittels maschinellem Lernen trainiert und liefern eine „confidence score“, inwieweit eine gegebene Antwort auf eine bestimmte Fragestellung zutrifft.

Nach dem Test und der Bereitstellung (Deployment) direkt im QnA Maker sind die QnA-KIPs prinzipiell über das Web ansprechbar – und können mithilfe des Microsoft Azure Bot Service auf verschiedenen Kanälen bereitgestellt und nach Wunsch auch mit Microsofts Language-Understanding-Dienst (LUIS) verknüpft werden, um (laut-)sprachbegabte KIPs zu erhalten.

3.2.8 Sprechende und physische KIPs

KIPs umfassen – wie eingangs erwähnt – nicht nur klassische Chatbots, mit denen man mittels Texteingabe und Displayausgabe (primär visuell) kommuniziert, sondern auch „sprachbegabte“ Varianten. Diese können mittels Mikrofon und Lautsprecher am herkömmlichen Computer genutzt werden, aber auch als Sprachassistent*innen auf mobilen Geräten (Smartphones, Tablets, Smartwatches und Wearables) oder „smarten“ Haushaltsgeräten (Smart TV, Smart Speaker) realisiert sein.

Die rein auditive (oder gemischt audio-visuelle) Kommunikation hat einige Besonderheiten und Vorteile: Sie eignet sich insbesondere für schnelle Antworten auf kurze Fragen, für „hands-free“-Szenarien (wie beim Auto- oder Radfahren oder anderen Aktivitäten, die nicht mit der Bedienung eines klassischen Mensch-Computer-Interface vereinbar sind). Zudem können spezielle Zielgruppen besser „angesprochen“ werden (z. B. Kinder, Senior*innen oder Menschen mit Beeinträchtigungen), und auditive Szenarien (etwa das Erlernen von gesprochenen Sprachen oder musikalischen Inhalten) können auf den spezifischen Sinneskanal fokussiert umgesetzt werden.

Aus technischer Perspektive ist dafür „nur“ ein weiteres Zwischenglied erforderlich; vor dem Erkennen der Inhalte einer menschlichen Äußerung (Sprachverstehen bzw. NLU) ist eine Spracherkennungs- bzw. Speech-to-Text-Funktion nötig, und umgekehrt muss aus der (in Form von Text hinterlegten) Reaktion des Systems eine sprachliche Antwort generiert werden.

Wie die Umsetzung einer (zusätzlichen) Sprech-Schnittstelle für Chatbots aussehen könnte, zeigen – in rudimentärer Form – z. B. die Startseiten von Pandorabots⁵³ oder das Chat-Widget des Bot-Entwicklungsdienstes Snatchbot⁵⁴: Dort kann man auch über Mikrofon und Lautsprecher ein einfaches Gespräch führen. Weitaus mächtiger sind jedoch Systeme, die auf Erkennung gesprochener Sprache (im Sinne von Speech Recognition oder Speech-to-Text) spezialisiert sind. Da sich einerseits die Chatbot-Plattformen primär im Umfeld der geschriebenen Sprache entwickelt haben bzw. der Tradition der Texteingabe entstammen und andererseits die Spracherkennung noch kein Standardwerkzeug der Web-Entwicklung ist⁵⁵, findet man eine Kombination von KIP und Spracherkennung vornehmlich bei den großen Playern (z. B. Amazon, Apple, Facebook, Google, IBM, Microsoft etc.). Der Einsatz kann durchaus auch im Baukastensystem erfolgen – so kann man eigene KIPs etwa mit Amazon Lex⁵⁶ erweitern oder die Spracherkennungsdienste von Microsoft⁵⁷ und Google⁵⁸ nutzen; üblicherweise greift man aber zumeist auf konkrete KIPs zurück, die bereits von den bekannten Unternehmen angeboten werden: auf

⁵³ <https://home.pandorabots.com/home.html>

⁵⁴ <https://snatchbot.me/>

⁵⁵ Ein vielversprechender Ansatz ist z. B. die Web Speech API (https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API), die aktuell allerdings nur sehr eingeschränkt einsetzbar ist.

⁵⁶ <https://aws.amazon.com/de/lex/>

⁵⁷ <https://azure.microsoft.com/en-us/services/cognitive-services/speech-to-text/>

⁵⁸ <https://cloud.google.com/speech-to-text?hl=de>

Amazons Alexa, Apples Siri, den Google Assistant oder Microsofts Cortana, die man um individuelle Fähigkeiten bereichern kann.

Actions für den Google Assistant

Der intelligente Sprachassistent von Google („Google Assistant“) ist vornehmlich auf Mobilgeräten mit Android-Betriebssystem und smarten Geräten (z. B. dem hauseigenen Smart Speaker „Home“) verfügbar und kann nicht nur Auskünfte erteilen, sondern auch verschiedene Aufträge ausführen (Smartphones und weitere Geräte steuern, einfache Aktionen in Apps durchführen etc.). Seit Ende 2016 ist es möglich, das Gesprächs- und Funktionsrepertoire des eigenen Google Assistant auch mit individuellen Inhalten zu erweitern, indem man ihn um sogenannte „Conversational Actions“ aus dem mittlerweile reichhaltigen Katalog⁵⁹ erweitert oder selbst welche erstellt.

Die Spracheingabe- und Sprachausgabefunktion wird von Google bereitgestellt, es müssen lediglich die einzelnen „Actions“ mit identifizierbaren Inhalten, ggf. auch mit einer Gesprächsstruktur und Funktionen befüllt werden; die in Text umgewandelten Spracheingaben müssen zuerst verstanden werden und danach (mit einer Sprachantwort oder Effekten wie beispielsweise der Steuerung von Webservices oder Geräten) beantwortet werden.

⁵⁹ <https://assistant.google.com/explore/>

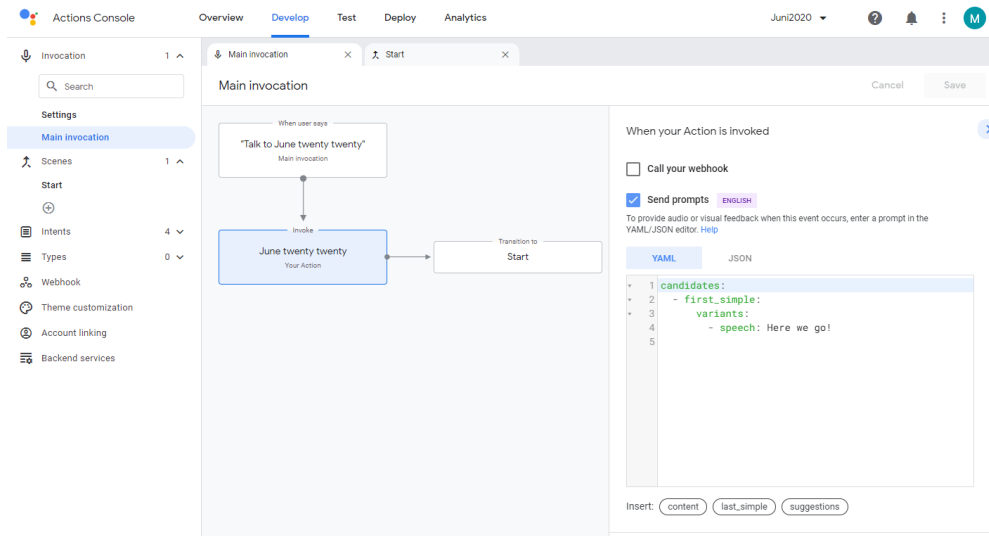


Abb. 14: Einblick in die neue, seit Juni 2020 erweiterte Google Actions Console (Stand vom 29. Juni 2020)

Dieser eigentliche Inhalt bzw. die „intelligente“ Komponente von Actions kann mithilfe des Actions SDK oder neuerdings auch direkt in der „Actions Console“ mit dem integrierten „Actions Builder“ gestaltet werden – vor Juni 2020 musste man dazu noch Dialogflow (s. o.) bemühen. Neben der inhaltlichen Gestaltung sind in der Actions Console noch weitere Angaben zu machen und Einstellungen vorzunehmen (z. B. die Einstellung, wie die Aktion aufgerufen wird, und Angaben, wie sie im Katalog erscheint usw.), um diese zu vervollständigen und im Katalog anbieten zu können. Dokumentation und Anleitungen zu Conversational Actions sind auf der Webseite⁶⁰ verfügbar; auf der übergeordneten Seite⁶¹ finden sich wei-

⁶⁰ <https://developers.google.com/assistant/conversational/overview>

⁶¹ <https://developers.google.com/assistant>

tere Informationen, wie man den Assistant beispielsweise mit seinen eigenen Apps verbindet oder wie man Web-Content aufbereiten kann, um ihn über den Assistant zugänglich zu machen.

Skills für Amazon Alexa

Analog zu Google stellt auch Amazon eine Reihe von Werkzeugen zur Verfügung, um dessen populäre Sprachassistentin „Alexa“ um weitere Fertigkeiten zu bereichern, die als „Skills“ bezeichnet werden. Ebenso wie beim Google Assistant verfügt Alexa, die etwa auf Amazons Echo-Gerätepalette haust, aber auch in andere Geräte integriert werden kann, über ein umfangreiches Repertoire an Konversationsmöglichkeiten, kann aber auch um individuelle Skills bereichert werden, die mit einer Startformel („Skill invocation“, z. B. „Alexa, starte den KIPH-Skill!“) aufgerufen werden kann. Amazon gruppiert die Skills in Smart Home Skills, Flash Briefing Skills (für informative Kurzinhalte), Video Skills und Music Skills, denen je ein spezifisches (vordefiniertes) Interaktionsmodell zugrunde liegt, sowie in Custom Skills, die selbst frei gestaltet werden können. Für die Gestaltung steht der „Amazon Skills Kit“ zur Verfügung – eine Art Werkzeugkasten mit unterschiedlichen Tools, Ressourcen und Anleitungen⁶². Ebenso wie bei Google erfolgt der Einstieg über die sogenannte Alexa Developer Console, für die ein Developer-Account erforderlich ist und wo man zum Einstieg beispielsweise einen einfachen Geburtstags-Skill mittels Vorlage erstellen, (z. B. naheliegenderweise auf Amazon Web Services) bereitstellen und diesen im Skills-Katalog veröffentlichen kann.

⁶² Einstieg und Informationen unter <https://developer.amazon.com/de-DE/alexa>

The screenshot shows the Alexa Developer Console interface for configuring a skill. The top navigation bar includes 'Your Skills', 'cake walk', 'Build', 'Code', 'Test', 'Distribution', 'Certification', 'Analytics', and 'Provide feedback'. The left sidebar contains a navigation menu with sections: 'CUSTOM' (Interaction Model, Utterance Conflicts (0), Invocation, Intents (5), Slot Types (3), JSON Editor), 'IN-SKILL PRODUCTS' (Interfaces, Endpoint, Intent History, Annotation Sets, Display), and 'ACCOUNT LINKING'. The main content area is titled 'Intents / CaptureBirthdayIntent' and shows 'Sample Utterances (6)'. A search box asks 'What might a user say to invoke this intent?'. Below it, six sample utterances are listed with slot markers: 'I was born in {month} {year}', 'I was born on {month} {day} {year}', 'I was born on {month} {day}', '{month} {year}', and '{month} {day} {year}'. A 'Dialog Delegation Strategy' dropdown is set to 'fallback to skill setting'. A notification at the top right states: 'Updates to sample utterances qualify for instant live updates. Learn more about live updates to your skill.'

Abb. 15: Geburtstags-Demo-Skill in der Alexa Developer Console (Stand vom 25. Mai 2020)

Weitere Lösungen: Siri, Cortana und Watson

Um individuelle Fertigkeiten kann auch Apples Sprachassistent*in „Siri“ bereichert werden, indem man solche einfach für sich selbst mit der Shortcuts-App⁶³ einrichtet oder mithilfe des „SiriKit“⁶⁴ entwickelt.

Auch Microsofts „Cortana“ bietet einen Katalog an verfügbaren Skills⁶⁵ und die Möglichkeit zur Entwicklung derselben⁶⁶. Die Erstellung erfolgt wiederum mithilfe des „Azure Bot Service“⁶⁷ und kann dort mit unterschiedlichen Komponenten aus Microsofts umfangreichen Cloud-Diensten (Azure Services) verknüpft werden. Auch hier bedingt die Vielfalt an Möglichkeiten ein gewisses Maß an Komplexität und Unübersichtlichkeit, wie sie etwa beim weitaus einfacheren „QnA Maker“ (s. o.) nicht gegeben sind.

In ähnlicher Weise wie bei Microsoft können auch aus dem umfangreichen Serviceangebot von IBMs künstlicher Intelligenz „Watson“⁶⁸ Sprachassistent*innen zusammengestellt werden, indem man den eigentlichen (individuell gestaltbaren) „Watson Assistant“ mit einer Speech-to-Text- und Text-to-Speech-Funktion ergänzt; die resultierenden KIPs können dann nicht nur auf den üblichen Interaktionskanälen bereitgestellt, sondern auch in geeignete Geräte integriert oder als „Voice Agent“ an das Telefonnetz angeschlossen werden, um Telefongespräche zu führen.⁶⁹

⁶³ <https://apps.apple.com/us/app/shortcuts/id915249334>

⁶⁴ <https://developer.apple.com/siri/>

⁶⁵ <https://www.microsoft.com/en-us/cortana/skills>

⁶⁶ <https://developer.microsoft.com/en-us/cortana/>

⁶⁷ <https://azure.microsoft.com/de-de/services/bot-service/>

⁶⁸ <https://www.ibm.com/watson>

⁶⁹ <https://cloud.ibm.com/docs/voice-agent>

Roboter

Unter der Voraussetzung, dass genügend finanzielle Ressourcen zur Verfügung stehen, können KIPs auch als physische Roboter realisiert werden. Relativ bekannt sind beispielsweise Pepper und NAO, zwei für Bildungssettings einsetzbare bzw. programmierbare Roboter von SoftBank Robotics⁷⁰, die auch eine prominente Rolle im erwähnten H.E.A.R.T.-Projekt⁷¹ spielen.

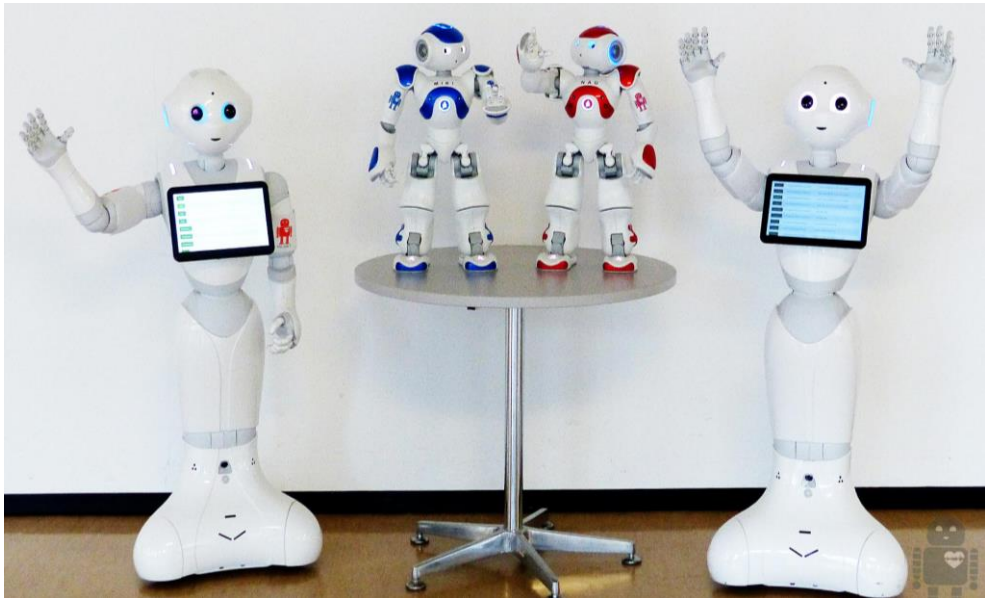


Abb. 16: Yuki, Miki, NAO und Pepper vom H.E.A.R.T.-Projekt
(Foto: „Forschungsobjekte“, Projekt H.E.A.R.T., CC BY 4.0⁷²)

⁷⁰ <https://www.softbankrobotics.com>

⁷¹ <https://www.project-heart.de/>

⁷² <https://www.project-heart.de/media/download/>

Auch Eigenentwicklungen sind denkbar und möglich, wie z. B. „Botter“ an der Universität Oberta de Catalunya illustriert (Torres Garrote, 2017) – der dort entwickelte Roboter-Prototyp soll den Lerneifer der Studierenden erhöhen, indem der jeweiligen Lernfortschritt mit lobenden oder anspornenden Reaktionen (in Form von Geräuschen, Licht oder Bewegung) beantwortet und dadurch dem Lernprozess eine besondere „Gefühlsqualität“ beigesteuert wird.

4 Ausblick

4.1 Vorteile und Potenziale

Die folgenden Einschätzungen haben sich im Verlauf der Bestandsaufnahme im KIPH-Projekt ergeben. Auch hier wird kein Anspruch auf Vollständigkeit und Unfehlbarkeit erhoben. Aus der Kombination von didaktischen Überlegungen und technischen Möglichkeiten lässt sich zunächst eine Reihe von Vorteilen zusammenfassen, die der Einsatz von KIPs an Hochschulen mit sich bringen kann.

- Die simulierte Interaktion und persönliche Ansprache, wie sie von KIPs im Gegensatz zu herkömmlichen Informationssystemen geleistet werden, kommt einem zentralen menschlichen Bedürfnis entgegen. Die Aufbereitung von Informationen, Inhalten und Materialien in Dialog- oder Gesprächsform eröffnet neue Möglichkeiten der Vermittlung, die herkömmliche (statische) Formate ergänzen oder bereichern können, die Adressat*innen zur Auseinandersetzung motivieren und helfen können, die Ziele und „Outcomes“ bestimmter Angebote besser zu erreichen.
- Mithilfe von KIPs werden Teile der individuellen Betreuung und Unterstützung von Zielgruppen (Mitarbeiter*innen, Studierende, Interessierte etc.) „skalierbar“. KIPs stehen als Service nicht nur zu jeder Tages- und Nachtzeit durchgängig (und mit derselben Stimmung) zur Verfügung, sondern kennen prinzipiell auch keine Kapazitätsgrenzen – sowohl bei der Anzahl ihrer Gesprächspartner*innen als auch bei der Betreuungsintensität. Somit sind auch für allgemeine Anlaufstellen, Groß(lehr)veranstaltungen oder beispielsweise MOOCs individuelle Auskünfte und Beratungen möglich, wie sie mit menschlichen Ressourcen kaum bereitgestellt werden könnten.

-
- Interaktionsmodalitäten, Inhalte und „Charakter“ von KIPs können prinzipiell auch auf persönliche Voraussetzungen, Bedürfnisse und Fortschritte der jeweiligen Adressat*innen zugeschnitten werden, was insbesondere bei Realisierung individueller Betreuungssettings (personalisiertes Lernen, Coaching etc.) zum Zug kommen kann.
 - Unabhängig davon unterstützen KIPs die Inklusion und Barrierefreiheit von Angeboten, indem sie bestimmte Zielgruppen als Medium besser unterstützen – wenn etwa Voice Assistants für Blinde und Sehbehinderte zur Verfügung gestellt werden oder mit KIPs eine entspanntere, auch als fehlertoleranter wahrgenommene Atmosphäre im Vergleich zur sozialen Interaktion mit Menschen (vgl. Winkler & Söllner, 2018, S. 24) gewählt werden kann.
 - Üblicherweise sind KIPs in populäre Kommunikationskanäle integriert, die bequem am Smartphone, aber auch am Desktop-Rechner erreicht werden können; die Nutzung erfordert keine gesonderte Infrastruktur mit speziellen Voraussetzungen. Die Interaktion mit KIPs gestaltet sich somit niederschwellig und kann problemlos in den Lern- und Arbeitsalltag integriert werden und auch als Instrument des „Seamless Learning“ gelten.
 - Konversationen mit KIPs werden üblicherweise gespeichert und können nachträglich analysiert werden; primärer Zweck ist wohl die Optimierung der Gesprächsfertigkeit von KIPs anhand von „realen“ Gesprächsdaten. Dieses Feature erweist sich aber auch darüber hinaus als praktisch: Feedback und Nutzungsstatistiken sind quasi in den Gesprächsverlauf „eingebaut“ und müssen nicht einer (ohnehin meist wenig aussagekräftigen) Logfileanalyse entnommen werden. Die Gesprächsdaten können aber auch Anknüpfungspunkte für Learning-Analytics-Projekte bieten, da sie Lernendendaten abseits der zentralen Plattformen (Lernmanagementsysteme) generieren.

4.2 Probleme und Herausforderungen

Neben den Vorzügen, die KIPs als neues Interaktions- und Vermittlungsmedium im Hochschulbereich mitbringen, sind beim Einsatz jedoch auch Fragestellungen zu berücksichtigen, die teils spezifisch für KIPs sind, teils aber auch generell mit der Digitalisierung von Settings zusammenhängen.

- Zunächst zu rechtlichen Aspekten: Nicht zuletzt durch die seit Mai 2018 geltende europäische Datenschutz-Grundverordnung ist ein sorgfältiger und bedachter Umgang mit personenbezogenen Daten verpflichtend – insbesondere bei KIPs, die solche Daten „verarbeiten“. Dies ist beispielsweise der Fall, wenn KIPs Fragen zur Person stellen und speichern, Daten aus dem Messenger-Account übernehmen, personenbezogene Daten mit externen Systemen austauschen oder User-Tracking und Profiling betreiben (was üblicherweise im Rahmen der Nutzungsstatistiken und für Analytics-Zwecke geschieht, aber nicht unbedingt Rückschlüsse auf konkrete Personen zulassen muss). Neben den eigentlichen Maßnahmen zum Datenschutz ist im gegebenen Fall noch weiteren Erfordernissen nachzukommen (Auskunftspflicht, Führen eines Datenverarbeitungsverzeichnisses, Abschluss eines Auftragsverarbeitungsvertrages bei Inanspruchnahme eines Services von Drittanbietern etc.). Weitere rechtliche Fragen betreffen den Einsatz in Lehrsettings, wenn etwa die Auseinandersetzung mit KIPs verpflichtender Bestandteil der Beurteilung ist.
- Neben rechtlichen Herausforderungen stellen sich auch eine Reihe ethischer Fragen, in erster Linie die nach der Transparenz: Inwiefern geben sich KIPs als solche zu erkennen? Diese – besonders im Zusammenhang mit der Rolle von Social Bots in der politischen Meinungsbildung erörterte – Frage betrifft nicht nur die Identität (z. B. in spektakulären Fällen wie der künstlichen Lehrassistentin Jill Watson), sondern auch und vor allem die Transparenz in Bezug auf eingesetzte Technologien. Die Nutzer*innen sollten zumindest über die grundlegende Funktionsweise der jeweiligen Lösung aufgeklärt werden und über die technischen Möglichkeiten und

Grenzen Bescheid wissen. Auch die Ziele des Einsatzes von KIPs sollten stets reflektiert, transparent kommuniziert und zur Diskussion gestellt werden, zumal dadurch unweigerlich neue Prozesse und neue Standards im Hochschulbetrieb etabliert werden. Schließlich sollte auch eine „Vision“ ernsthaft diskutiert werden, die im Zusammenhang mit Automatisierungs- und Digitalisierungsprozessen aufkommt: Können und sollen KIPs Lehrpersonen ersetzen? Werden menschliche Lehrer*innen dadurch obsolet? Abseits von diffusen Befürchtungen ist es erforderlich, ein realistisches Bild von den Möglichkeiten und Grenzen und eventuell auch Vorteilen zu zeichnen (vgl. etwa Edwards & Cheok, 2018 in Hinblick auf die Kompensation von Lehrer*innenmangel).

- In „technischer“ Hinsicht sei zunächst noch einmal ein Begleitumstand der Entscheidung für eine bestimmte Lösung angesprochen, nämlich die Abhängigkeit von den Anbieter*innen: Meist handelt es sich im Bereich von Software-as-a-Service-Angeboten (also Web- oder Cloud-basierten Entwicklungsplattformen) um proprietäre, geschlossene Systeme, und viele Chatbot-Entwicklungsplattformen sind nur als Cloud-Service verfügbar. Das betrifft nicht nur Fragen der Transparenz (Inwieweit ist die Funktionsweise einsehbar? Was geschieht mit den Daten? Ist der Quellcode des Systems Open Source?), sondern auch das Problem, dass die eingegebenen Daten und Inhalte oft nicht (bzw. nicht in standardisierten, offenen Formaten) exportierbar und in anderen Kontexten wiederverwendbar sind. Ein „Lock in“ (also der Umstand, dass die gewählte Lösung beibehalten werden muss) stimmt vor allem angesichts des fraglichen langfristigen Fortbestands von (vor allem kleineren) Anbieter*innen bedenklich. Weitere technische Herausforderungen betreffen die Interoperabilität einzelner Bestandteile eines KIP-Systems – etwa, wenn KIPs aus heterogenen Komponenten zusammengestellt sind, auf Daten in anderen Systemen zugreifen sollen oder auf mehreren Plattformen oder Kommunikationskanälen operieren, die unterschiedliche Voraussetzungen mitbringen. Derlei Anforderungen sollten von Anfang an (vor der Auswahl eines konkreten

Systems) mitbedacht werden, da sie – wenn sie nicht ohnehin vom gewählten Angebot von Haus aus unterstützt werden – erheblichen Entwicklungsaufwand bedeuten können.

- Für zukunftsorientiertere Szenarien (intelligente, selbstlernende künstliche Akteur*innen, etwa als vorausschauende Lehrassistent*innen oder einfühlsame Mentor*innen eingesetzt) sollte man sich jedenfalls ein realistisches Bild von den Möglichkeiten und Grenzen der zeitgenössischen „Künstlichen Intelligenz“ machen. Gegen die jüngsten Erfolge bei Mensch-Maschine-Duellen (bei Brettspielen wie Go oder Echtzeit-Strategiespielen wie Dota 2) oder bei der Simulation menschlichen Gesprächsverhaltens (Blender und Meena) können nicht nur prinzipielle philosophische Bedenken ins Feld geführt werden (Locus classicus ist wohl Dreyfus, 1972), sondern auch empirische Beispiele wie etwa das Tay-Fiasko⁷³ und prinzipielle Probleme des Machine Learning, etwa die Undurchsichtigkeit und Manipulierbarkeit von komplexen Algorithmen, die beispielsweise zur – sichtbaren oder unsichtbaren – Benachteiligung bestimmter Gruppen führen können.
- Doch zurück zu bescheideneren Vorhaben: Auch bezogen auf die Entscheidung, ob man KIPs einsetzen möchte oder sollte, stellt sich die Frage nach dem Verhältnis von Aufwand und erwartbarem Nutzen – ein Problem, das nicht nur KIPs im Speziellen, sondern generell die Digitalisierung komplexerer (akademischer) Praxis betrifft. Man sollte sich im Klaren darüber sein, dass „performante“ KIPs in allen Phasen (Planung, Erstellung, Einsatz, Verbesserung) ein hohes Ausmaß an Bemühungen und Ressourcen erfordern und dass insbesondere in der Planungsphase genau abgewogen werden sollte, welche Ziele mit der Entwicklung von KIPs erreicht – und besser als mit herkömmlichen oder alternativen Lösungen erreicht – werden sollen. Im Hinblick auf die Akzeptanz und nachhaltige Veranke-

⁷³ [https://de.wikipedia.org/wiki/Tay_\(Bot\)](https://de.wikipedia.org/wiki/Tay_(Bot))

rung ist zu bedenken, dass KIPs anfangs aufgrund des neuen technologischen Zugangs populär sein könnten, aber kein nachhaltiges Medium darstellen („novelty effect“, vgl. Winkler & Söllner, 2018, S. 17). Das anfängliche Interesse könnte einerseits nach einer Gewöhnungsphase bzw. im Zuge der allgemeinen Etablierung von KIPs abflauen, andererseits könnte der spielerische Zugang von den vermittelten Inhalten ablenken und so die eigentliche Intention konterkarieren. Gerade in der ersten Versuchs- und Experimentierphase des Einsatzes innovativer Technologien sollten auch derartige Effekte in die Evaluation miteinbezogen werden.

- Damit verbunden sind auch gesellschaftliche und kulturelle Fragestellungen: Werden beispielsweise junge Menschen und Studierende durch digitale Assistent*innen mit all ihren Annehmlichkeiten (Terminmanagement, Wissensverfügbarkeit, Empfehlungswesen, Ablauf-Unterstützung) zur Unmündigkeit erzogen bzw. teilweise sogar entmündigt? Opfert man der Bequemlichkeit eines einfachen Frage-Antwort-Dialogs unvorhergesehene Erkenntnisse und einen umfassenden Wissenskontext, wie sie für das Studium von Texten charakteristisch sind? Leisten KIPs einem trivialen Kommunikationsstil, verarmter Sprache (kurzen Sätzen, reduziertem Vokabular) und profanen Inhalten Vorschub (vgl. Winkler & Söllner, 2018, S. 20 und 26 sowie Kunze, 2019)? Kann man – insbesondere im aufgeklärten akademischen Umfeld – eine „angemessene“ und aufgeklärte Erwartungshaltung gegenüber KIPs erwarten – sollen diese als echte (wenn auch künstliche) Persönlichkeiten konzipiert und behandelt werden oder explizit nur als einfache Werkzeuge mit wohldefinierten, aber beschränkten Interaktionsmöglichkeiten?⁷⁴ Die konkrete Ausgestaltung und der Grad an Komplexität und Menschlichkeit von KIPs stellen sich vor diesem Hinter-

⁷⁴ Vgl. Soelzer, 2016 sowie Lloyd, 2016, die angesichts überzogener Erwartungen an deren Intelligenz vorschlägt, Bots nicht „humanoid“ zu gestalten, sondern „mechanomorph“, also maschinenähnlich, und die Erwartungshaltung und „Ästhetik“ entsprechend herunterzuschrauben.

grund nicht nur als bloße Designfragen heraus, sondern als Facette einer übergreifenden Fragestellung, die das „Zusammenleben“ von Mensch und IKT-Systemen in einer zunehmend technologisch durchdrungenen Welt betrifft.

4.3 Lessons für die Praxis

Wie nun beginnen? Aus der Beschreibung einzelner Einsatzszenarien und der Vorstellung einzelner Werkzeuge sollten die wesentlichen Kriterien für die Auswahl eines Systems und Grundzüge des weiteren Vorgehens ersichtlich sein. Für konkrete Vorhaben ist eine intensivere Auseinandersetzung mit in Frage kommenden Entwicklungsplattformen wohl unabdingbar, ebenso gründliche Praxistests und die laufende Weiterentwicklung des Resultats.

Der Leitfaden schließt mit einer Reihe allgemeiner Erkenntnisse, die im Rahmen der Arbeitsgruppe notiert wurden und gewissermaßen das Fazit der Auseinandersetzung mit KIPs darstellen.

- Grundsätzlich scheint die Umsetzung von Hochschul-KIPs aktuell am ehesten im organisatorischen Bereich aussichtsreich – wenn es primär um die Automatisierung bestimmter Services (wie allgemeine Beratungsleistungen oder Auskünfte) geht, die sich weitaus anders gestaltet als etwa die Schaffung einer intelligenten künstlichen Persönlichkeit. Für ausgefeilte didaktische Settings mit solchen KIPs ist einerseits der Gesamtaufwand sehr hoch, andererseits sind die dazu erforderlichen, über das Standardrepertoire von Bot-Entwicklungsplattformen hinausgehenden Technologien (z. B. Learning-Analytics-Systeme, Machine-Learning-Anwendungen, funktionierende APIs und Schnittstellen zu Lernmanagementsystemen etc.) entweder noch nicht weit gediehen oder kaum für Nicht-Spezialist*innen verfügbar bzw. niederschwellig genug. Ambitioniertere Vorhaben scheinen am ehesten in größeren (und großzügig ausgestatteten) Entwicklungsprojekten realisierbar; für erste Schritte sollten eher punktuelle Einsatzszenarien oder einfache Anwendungen für den Hochschulbetrieb erwogen werden.

-
- Demgegenüber ist jedoch zu betonen, dass bei geeigneten Szenarien und gutem didaktischen Design KIPs auch ohne technische Feinassen gut funktionieren können – beispielsweise KIPs, die gänzlich ohne Intent-Erkennung auskommen, sondern nur durchgeplante Gesprächsverläufe mit vorwiegend vorgegebenen Antworten bieten, oder einfach erstellte Supportangebote wie mithilfe von Question-and-Answer-Modulen realisierte Auskunfts-KIPs, die man besonders in Ausnahmesituationen – wie die zur Zeit der Abfassung anhaltende COVID-19-bedingte Krisensituation – zu schätzen lernt. Entscheidend ist weiters eine gute Einbettung in die jeweiligen Prozesse (Lehr-Lernumgebungen, praktische Zusammenhänge, Verwaltungsabläufe etc.). KIPs funktionieren nur, wenn sie in den jeweiligen Prozess integriert und direkt „vor Ort“, zeitnah und einfach zugänglich sind, also auch abseits vom klassischen Lern- und Arbeitsplatzrechner z. B. mit dem eigenen Smartphone oder über lokale Lösungen (Kiosksysteme, Voice Assistants, Roboter) in Anspruch genommen werden können.
 - Generell sollte bei einem Vorhaben zur Umsetzung von KIPs der Aufwand nicht zu niedrig veranschlagt werden. Die einfache Bedienbarkeit und Übersichtlichkeit einiger Entwicklungsplattformen könnten im ersten Moment darüber hinwegtäuschen, dass nicht nur Ressourcen für die Erstellung und Wartung des KIP erforderlich sind. Der eigentliche Aufwand liegt – eine ausgereifte Lösung vorausgesetzt – meist weniger auf der „technischen“ Seite, sondern im Design von KIPs, das u. a. das allgemeine Setup sowie die Planung von Konversationslogik, Sprache und Persönlichkeit der KIPs betrifft (vgl. z. B. Mercier, 2017). Ein wichtiger Aspekt für „plastischer“ gestaltete KIP ist das Konversationsdesign, das quasi das „Erscheinungsbild“ des Produkts betrifft und mit einfachen Flowcharts sicherlich nicht erschöpfend geplant werden kann, sondern zumindest ein Storyboard oder elaborierte Designmethoden erfordert. Darüber hinaus ist es mit der Veröffentlichung einer Version 1.0 nicht getan; KIPs müssen vielmehr kontinuierlich weiterentwickelt werden (vgl. etwa den Hollander et al., 2019; Shevat, 2017), was etwa das „Debugging“ missglückter Gesprächs-

verläufe oder falscher Intent-Zuordnungen, die Feinabstimmung anhand von protokollierten (realen) Gesprächsverläufen und auch das Einpflegen neuer Informationen umfasst.

- In Bezug auf (fremde und auch eigene) Qualitätsansprüche stellt sich das Problem, dass KIPs vermutlich am Niveau der Performance von Alexa und Co. gemessen werden – ähnlich wie bei der Ästhetik und Qualität professioneller Videos, die im normalen Hochschulumfeld mit den zur Verfügung stehenden Ressourcen kaum oder nur in Einzelfällen erreichbar ist. Eine mögliche Abhilfe besteht darin, Studierende aktiv zu beteiligen und ein Problembewusstsein entwickeln zu lassen, indem sie selbst KIPs (mit-)gestalten. Die Frage, ob Studierende mithilfe von KIPs noch weiter serviert und an die Hand genommen oder vielmehr zur eigenständigen Auseinandersetzung und Reflexion angeleitet werden sollen, betrifft auch die oben angesprochene aufgeklärte Haltung und sollte in der akademischen Auseinandersetzung jedenfalls diskutiert werden. In diesem Umfeld scheint eine bloße (für alltägliche KIP-Einsatzszenarien charakteristische) Kund*innen- und Konsument*innen-Haltung sowie der Fokus auf Entertainment (bestenfalls Edutainment) unangemessen – und dem widerspricht nicht, dass der Befassung mit KIP ein spielerisches Element inhärent ist.
- Was den didaktischen Einsatz von KIP an Hochschulen betrifft, kann man aktuell eher von „Bots in Education“ statt von „Educational Bots“ sprechen – KIPs können derzeit spezielle Teilaufgaben im Lehrbetrieb übernehmen, aber autonom vermutlich (noch?) keine durchwegs befriedigende Lehre durchführen oder gar Lehrpersonen ersetzen. Die Unterstützung von Lehrenden und Entlastung bei den administrativen und organisatorischen Agenden scheinen aktuell vielversprechende Ausgangspunkte für den Einsatz von KIPs zu sein... wenn man das Experiment wagen will, wozu der Leitfaden hoffentlich ermutigt hat. Viel Erfolg beim Ausprobieren und Erstellen von KIPs!

5 Literaturverzeichnis

Berger, R., Ebner, M. & Ebner, M. (2019). Conception of a Conversational Interface to Provide a Guided Search of Study Related Data. *International Journal of Emerging Technologies in Learning*, 14(7), 37-47.

<https://doi.org/10.3991/ijet.v14i07.10137>

Brown, M. et al. (2020). 2020 EDUCAUSE Horizon Report. *Teaching and Learning Edition*. EDUCAUSE.

<https://library.educause.edu/resources/2020/3/2020-educause-horizon-report-teaching-and-learning-edition>, Stand vom 26. Juni 2020.

Carstensen, N. & Roedenbeck, M. (2019). Chatbots in der Studienorientierung. In T. Barton, C. Müller, & C. Seel (Hrsg.), *Hochschulen in Zeiten der Digitalisierung: Lehre, Forschung und Organisation* (S. 27–40). Springer Fachmedien.

https://doi.org/10.1007/978-3-658-26618-9_3

Clark, D. (12. Dezember 2017). *Invisible LMS: the LMS is not dead, but it needs to be invisible – front it with a chatbot*. Donald Clark Plan B.

<http://donaldclarkplanb.blogspot.com/2017/12/invisible-lms-lms-is-not-dead-but-it.html>, Stand vom 5. März 2020.

den Hollander, K., Hulsebosch, J. & van Luipen, S. (2019). *Meet Beau: A reflection chatbot*. <http://www.joitskehulsebosch.nl/wp-content/uploads/2019/03/Meet-Beau-the-reflection-chatbot-light-version.pdf>,

Stand vom 19. Juni 2020.

Dreyfus, H. (1972). *What Computers Can't Do: The Limits of Artificial Intelligence*. Harper & Row.

Edwards, B. I. & Cheok, A. D. (2018). Why Not Robot Teachers: Artificial Intelligence for Addressing Teacher Shortage. *Applied Artificial Intelligence*, 32(4), 345–360. <https://doi.org/10.1080/08839514.2018.1464286>

Goel, A. (2016). *A teaching assistant named Jill Watson*.

<https://www.youtube.com/watch?v=WbCgUlCylTA>

-
- Grams, B.** (2019). Künstliche Intelligenz in der Coaching-Praxis. *Coaching-Magazin*, 4 | 2019. <https://www.coaching-magazin.de/beruf-coach/kuenstliche-intelligenz-in-der-coaching-praxis>, Stand vom 26. Juni 2020.
- Hahn, R.** (2019). *Mit Chatbots zum Invisible LMS*. Krems. <http://webthesis.donau-uni.ac.at/thesen/99094.pdf>, Stand vom 15. Januar 2020.
- Handke, J.** (2020). *Humanoide Roboter: Showcase, Partner und Werkzeug*. Baden-Baden: Tectum. <https://doi.org/10.5771/9783828871359>
- Kowald, C. & Bruns, B.** (2019). New Learning Scenarios with Chatbots – Conversational Learning with Jix: From Digital Tutors to Serious Interactive Fiction Games. *International Journal of Advanced Corporate Learning*, 12(2), 59–62.
- Kunze, L.** (8. November 2019). *How to deprogram homicidal sex robots*. <https://www.youtube.com/watch?v=0mupm59Zpls>
- Krommer, A.** (16. April 2018). Warum der Grundsatz „Pädagogik vor Technik“ bestenfalls trivial ist. *Bildung unter Bedingungen der Digitalität*. <https://axelkrommer.com/2018/04/16/warum-der-grundsatz-paedagogik-vor-technik-bestenfalls-trivial-ist/>, Stand vom 7. Juli 2020.
- Lloyd, A.** (25. März 2016). Mechanomorphs and the politeness of machines. *Medium*. <https://medium.com/@alexislloyd/mechanomorphs-and-the-politeness-of-machines-be69fd1c7d6b>, Stand vom 26. Juni 2020.
- McElvaney, P.** (21. Mai 2018). *10 Secrets To Transforming L&D With A Chatbot*. ELearning Industry. <https://elearningindustry.com/transforming-l-d-with-a-chatbot-10-secrets>, Stand vom 5. März 2020.
- Mercier, M.** (4. Juni 2017). Are You Making Any Of The Following Mistakes When Creating Chatbots? *Botpress Blog*. <https://botpress.com/blog/are-you-making-any-of-the-following-mistakes-when-creating-chatbots>, Stand vom 26. Juni 2020.
- mmb Institut** (2019). *Mmb Learning Delphi 2018/2019 – Auf dem Weg zum Assisted Learning?* https://www.mmb-institut.de/wp-content/uploads/mmb-Trendmonitor_2018-2019.pdf, Stand vom 4. Juni 2020.
- Niebuhr, C.** (21. September 2018). Chatten gegen Lernstress. *MERTON Magazin*. <https://merton-magazin.de/chatten-gegen-lernstress>, Stand vom 30. Juli 2019.

- Shevat, A.** (2017). *Designing Bots: Creating conversational experiences*. O'Reilly Media.
- Soelzer, E.** (25. März 2016). Where many bots will fail. *Medium*. <https://blog.howdy.ai/where-many-bots-will-fail-68ae163e2473>, Stand vom 26. Juni 2020.
- Torres Garrote, A.** (2017). *The UOC designs a robot to enhance the digital learning experience*. <https://www.uoc.edu/portal/en/news/actualitat/2017/259-prototype.html>, Stand vom 22. Juni 2020.
- White, A.** (13. März 2019). How AI Mentors are the Next Step in Learning. *Medium*. <https://chatbotnewsdaily.com/how-ai-mentors-are-the-next-step-in-learning-4c8f250ac6ef>, Stand vom 9. Juli 2020.
- Winkler, R. & Söllner, M.** (2018). *Unleashing the Potential of Chatbots in Education: A State-Of-The-Art Analysis*. Academy of Management Annual Meeting (AOM), Chicago, USA. <https://www.alexandria.unisg.ch/254848/>, Stand vom 5. März 2020.
- Wittke, A.** (14. März 2017). Vortrag „Bots in der Lehre“ an der RWTH Aachen. *Online By Nature*. <http://www.onlinebynature.com/2017/03/vortrag-bots-in-der-lehre-an-der-rwth-aachen/>, Stand vom 22. Juni 2020.
- Zawacki-Richter, O., Marín, V. I., Bond, M. & Gouverneur, F.** (2019). Systematic review of research on artificial intelligence applications in higher education – where are the educators? *International Journal of Educational Technology in Higher Education*, 16(39). <https://doi.org/10.1186/s41239-019-0171-0>



**Verein Forum Neue Medien
in der Lehre Austria**
Liebiggasse 9/II
A-8010 Graz
Tel. +43 660 5948 774
Mail: office@fnma.at
Web: www.fnma.at

