

Vorlesung

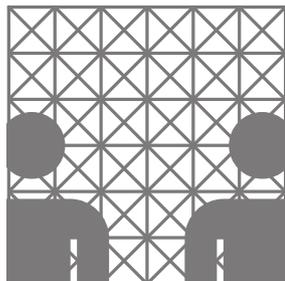
Sprachdialogsysteme

Timo Baumann
baumann@informatik.uni-hamburg.de



<https://nats-www.informatik.uni-hamburg.de/SDS20>

Universität Hamburg, Department of Informatics
Language Technology Group



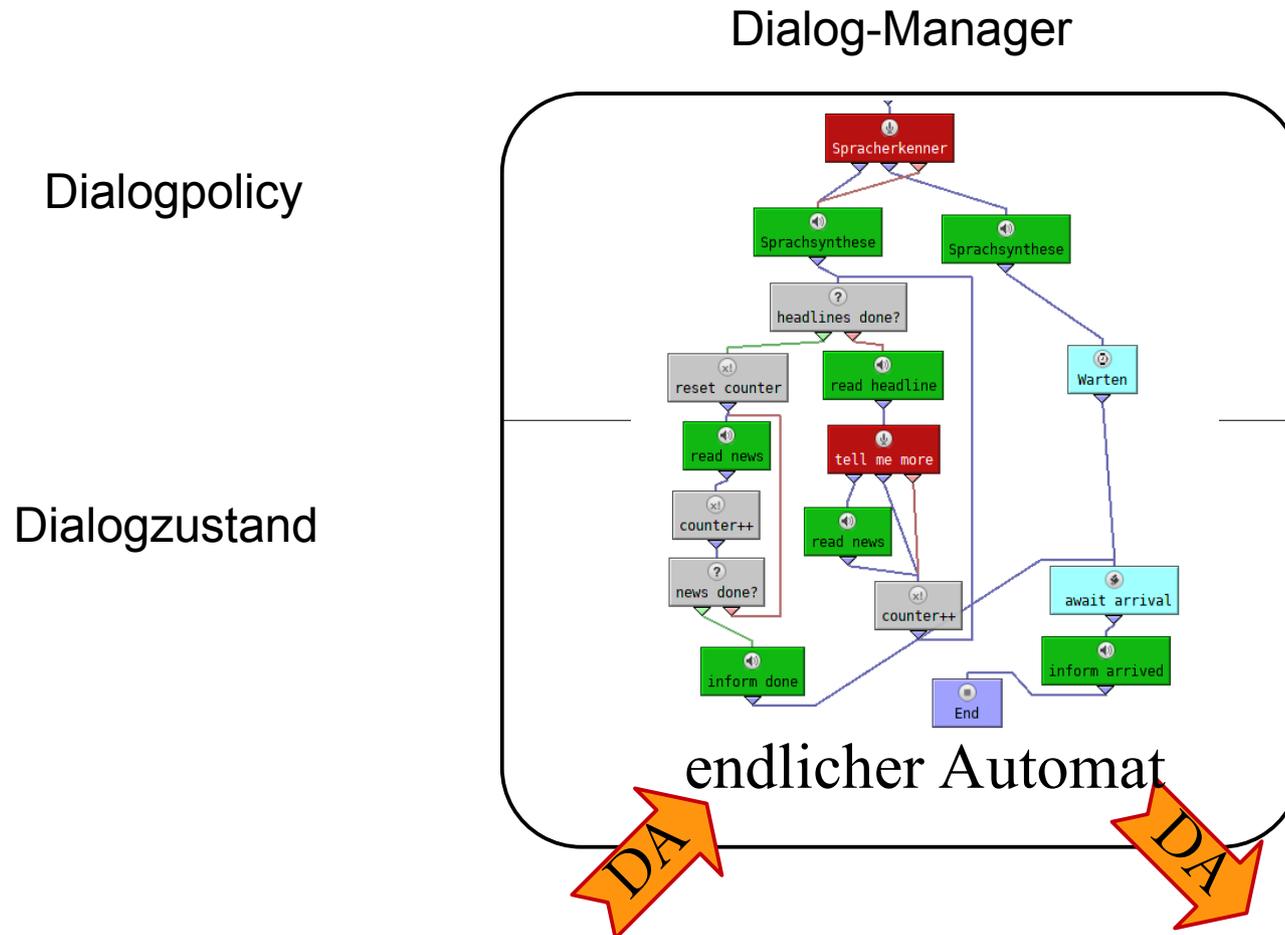
Heute

Reprise zum Frame-basierten Dialogmanagement

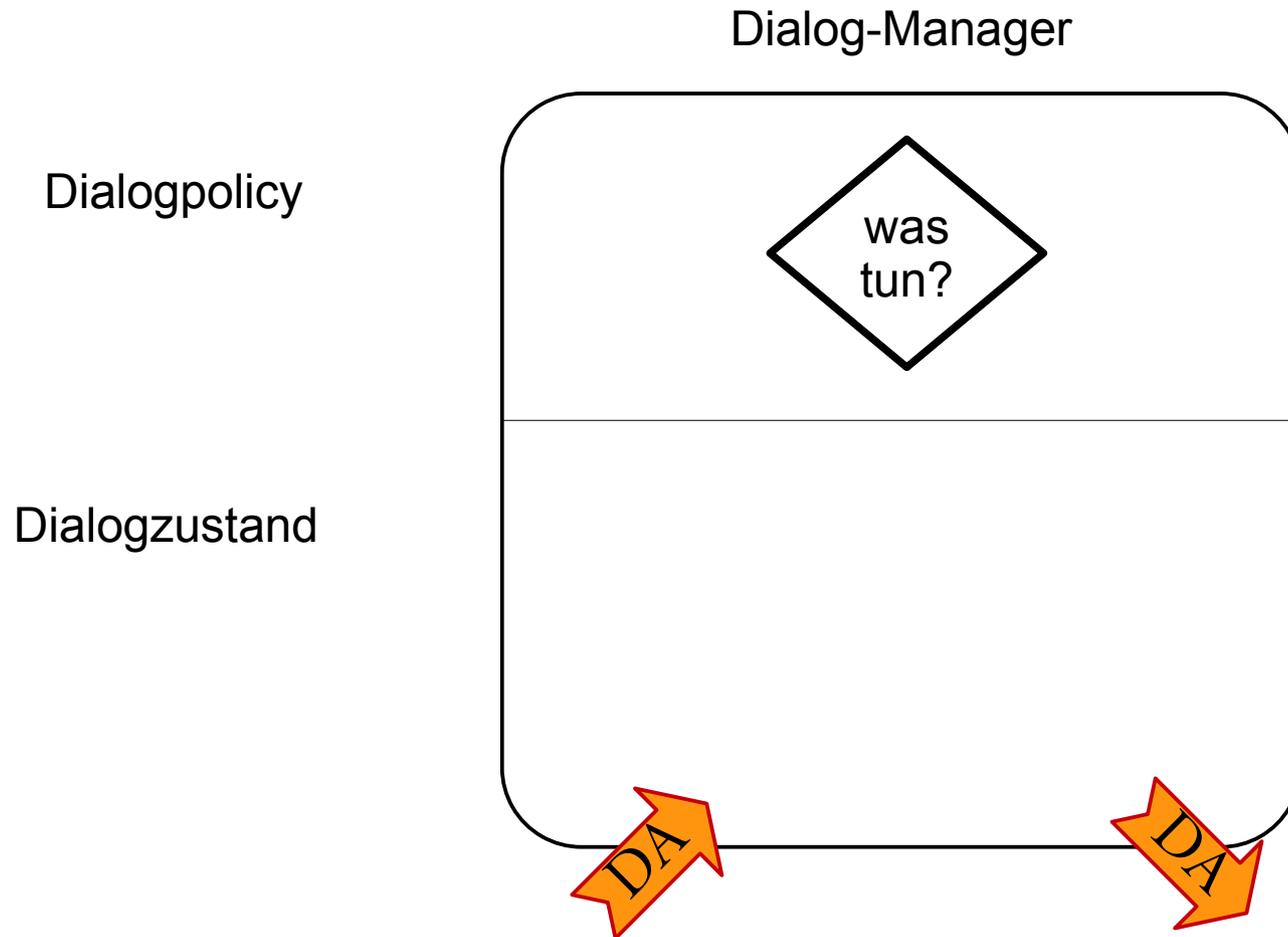
Unsicheres Wissen und probabilistisches Dialogmanagement

- Auswahl bester nächster Aktionen: MDP
- Modellierung von Zustandsalternativen: POMDP

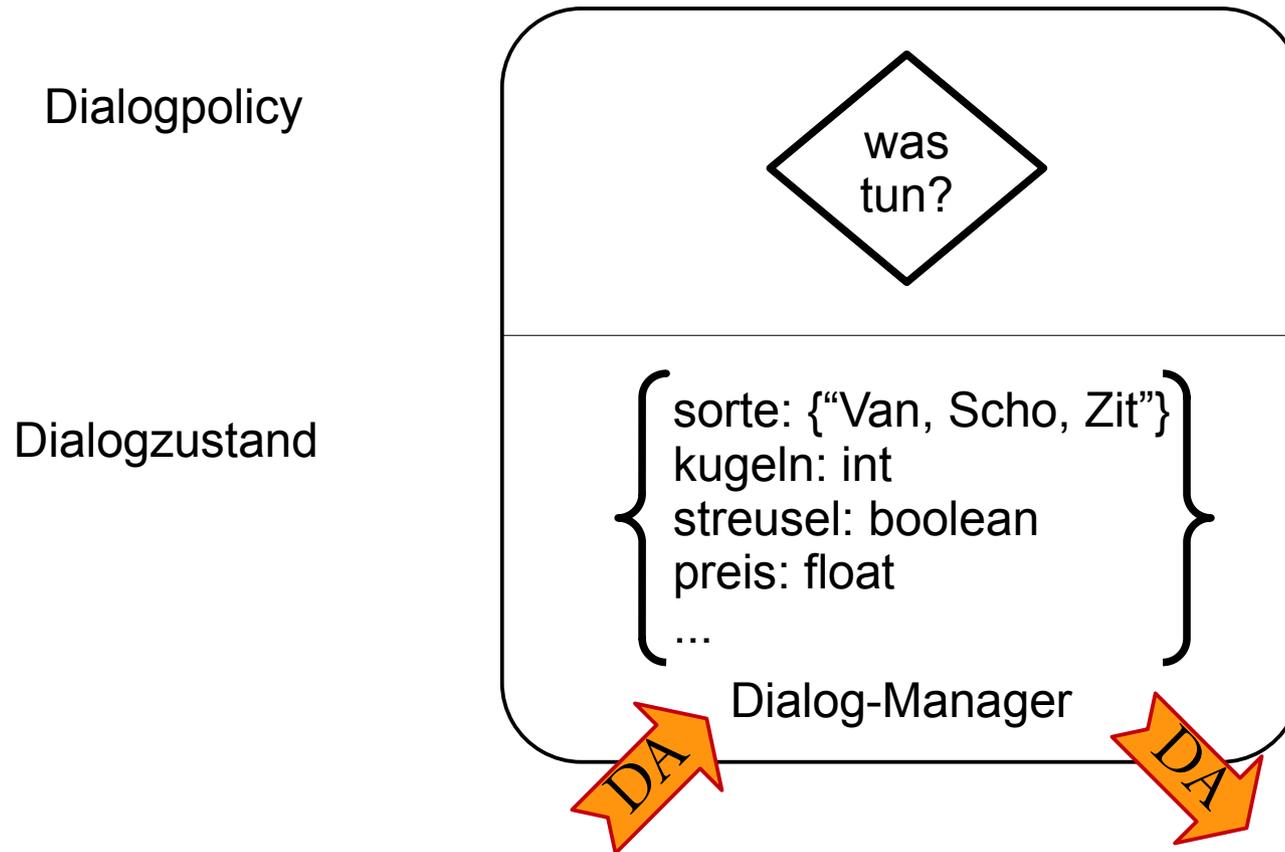
Komponenten im Dialogmanager



Komponenten im Dialogmanager



Komponenten im Dialogmanager: Frame-basiertes Management



Frame-Semantik

Dialog-Zustand:

- bestimmt aktuell relevanten Frame (oder mögliche Frames)
- bestimmt Ausgabe-Frame basierend auf Eingabe-Frame
- zum Beispiel: [kaufen : [Verkäufer: ich ; Käufer: Anrufer ; Ware : ? ; Preis : ?]]

Eingabeseite:

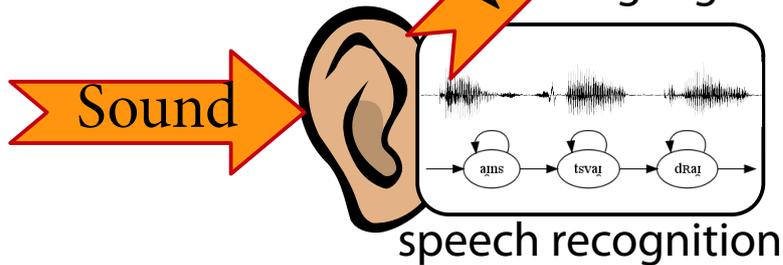
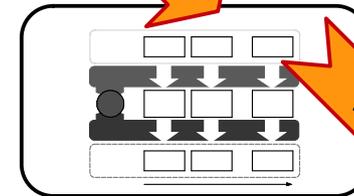
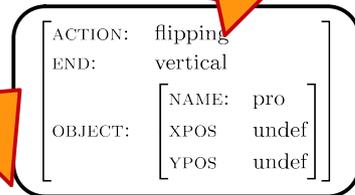
- Äußerung → Frame-Inhalt
“Hallo, ich möchte
3 Kugeln mit Sahne.”

Ausgabeseite:

- Frame-Inhalt → Äußerung
“Okay.
Das kostet 4 Euro 40.”

Ein einfacher Dialogagent

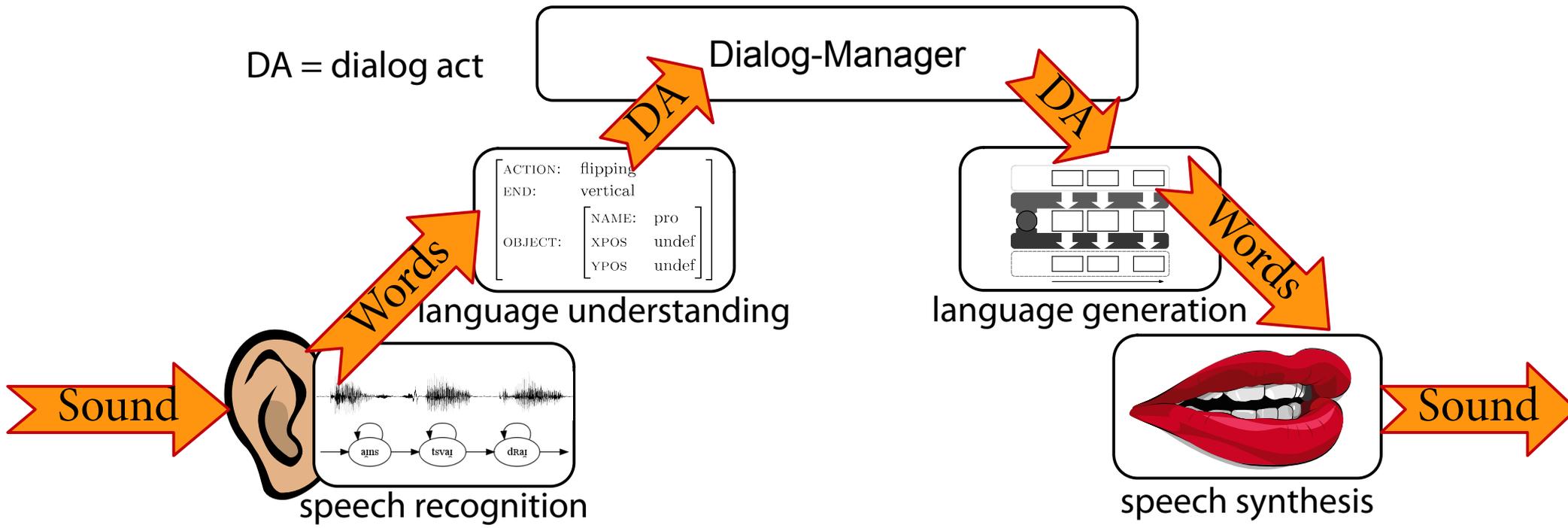
DA = dialog act



language generation



speech synthesis



Vorteile/Nachteile

endlicher Automat

Frame-basiert

FSM-basiertes DM

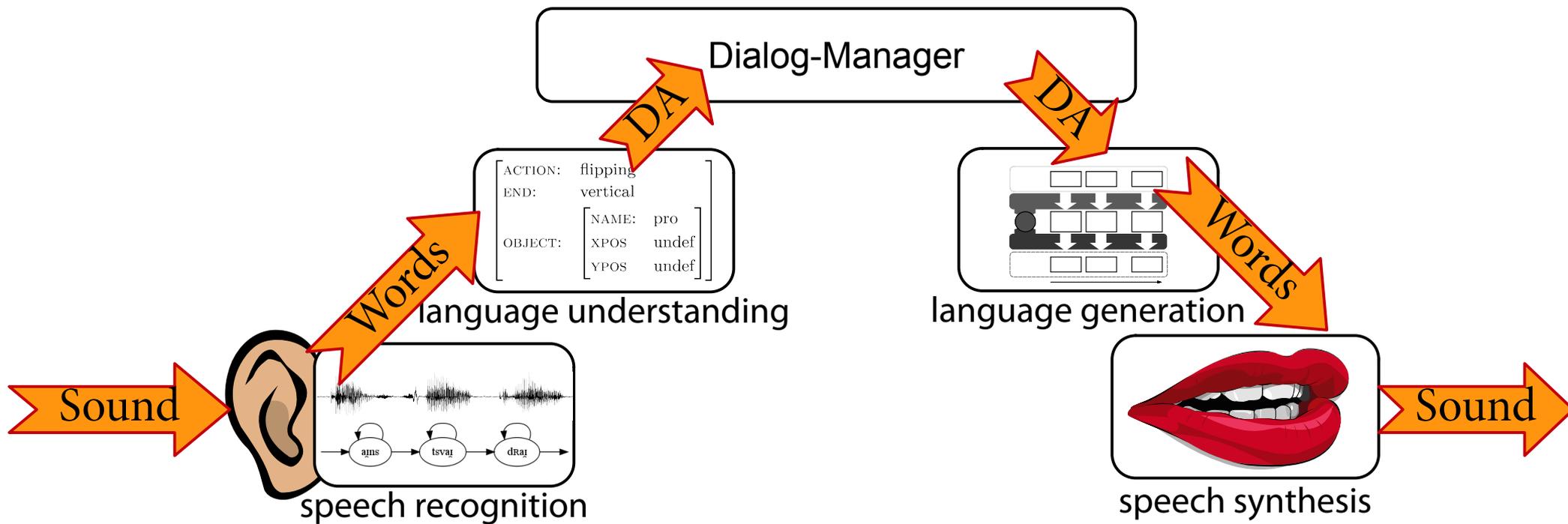
- spezialisierte State-Machine (kann auf Spezifika der Dialogdomäne eingehen)
- aufwendige Fallunterscheidungen
- Vorhersehen des Nutzerverhaltens

Frame-basiertes DM

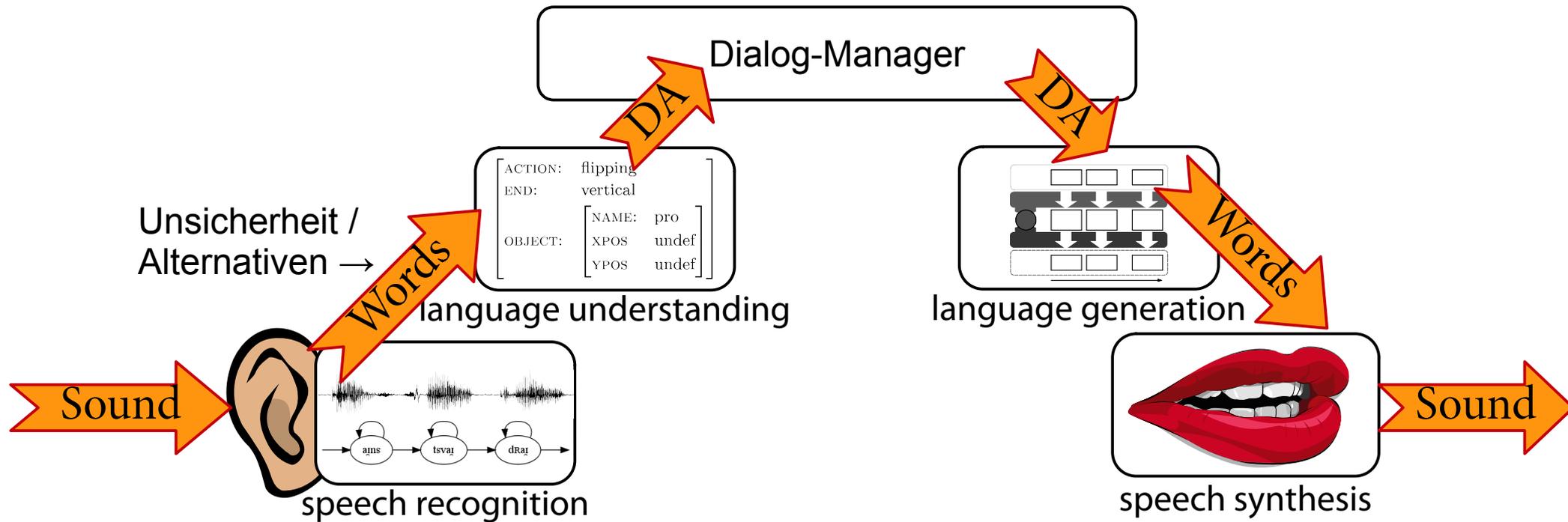
- generische State-Machine (aufwendig zu erstellen, dann aber sehr praktisch)
- keine fachspezifischen Abweichungen möglich

- Unsicherheit die sich aus möglichen Verarbeitungsfehlern ergibt
- kein Umgang mit Vagheit sprachlicher Äußerungen
- keine Anpassung an den Nutzer
- kein "Lernen" nützlichen Verhaltens

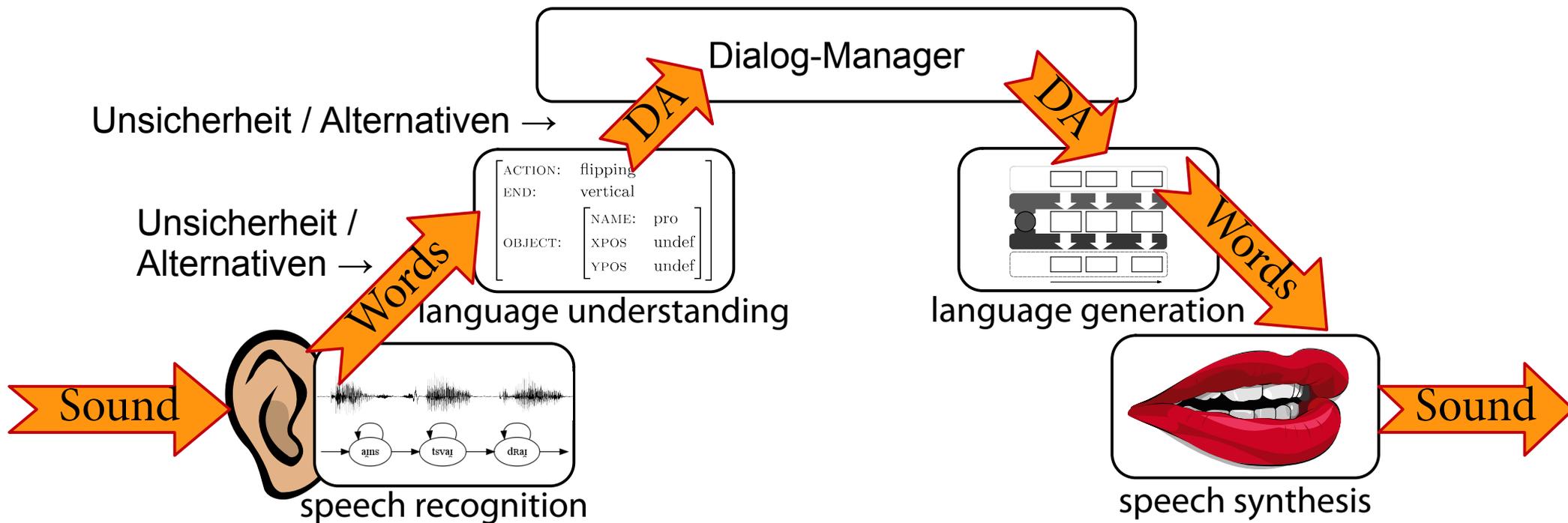
Ein probabilistischer Dialogagent



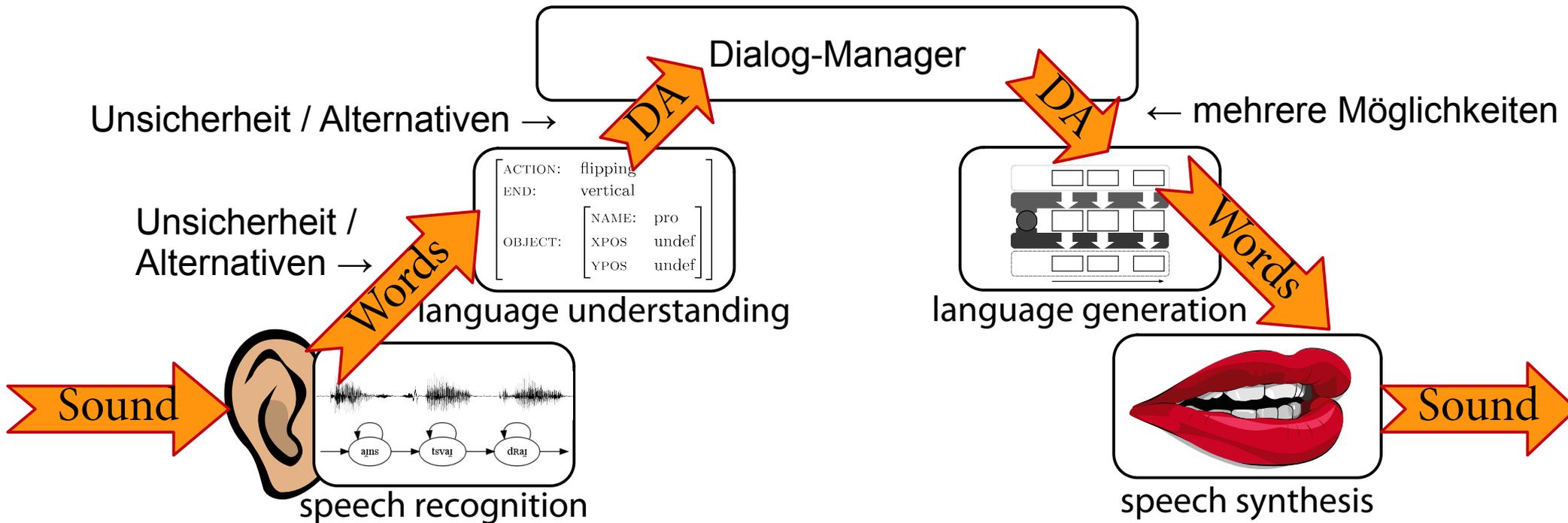
Ein probabilistischer Dialogagent



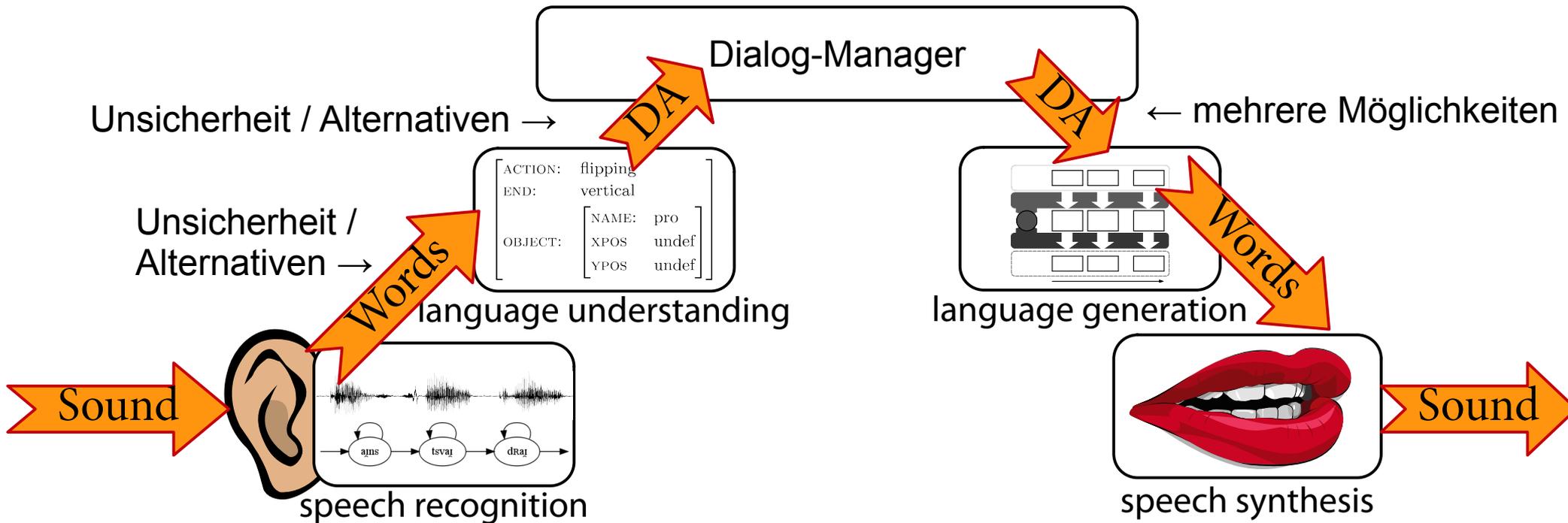
Ein probabilistischer Dialogagent



Ein probabilistischer Dialogagent



Ein probabilistischer Dialogagent



Ziel: "beste" Möglichkeit auswählen

no more ice cream

- Beispiel Fahrkartenverkauf
- S: “Nennen Sie bitte Ihren Abfahrtsort.”
U: “ Hamburg”
- ASR?
- NLU?
- System?

no more ice cream

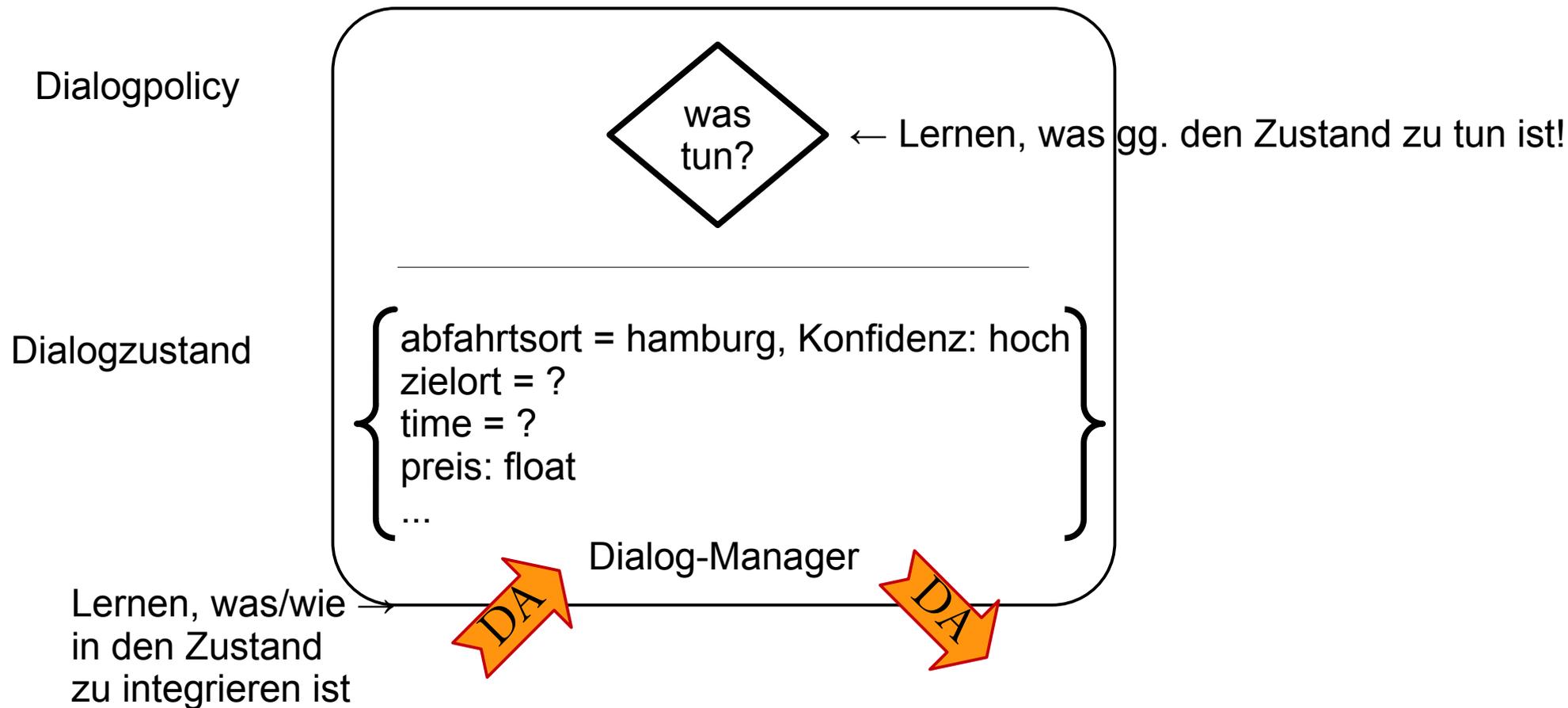
- Beispiel Fahrkartenverkauf
- S: “Nennen Sie bitte Ihren Abfahrtsort.”
U: “ Hamburg”
- ASR?
- NLU?
- System?

no more ice cream

- Beispiel Fahrkartenverkauf
- S: “Nennen Sie bitte Ihren Abfahrtsort.”
U: “ Hamburg”
- ASR?
- NLU?
- System?

Konfidenz: wie sicher bin ich mir bei meiner Aussage.

Probabilistisches Dialogmanagement



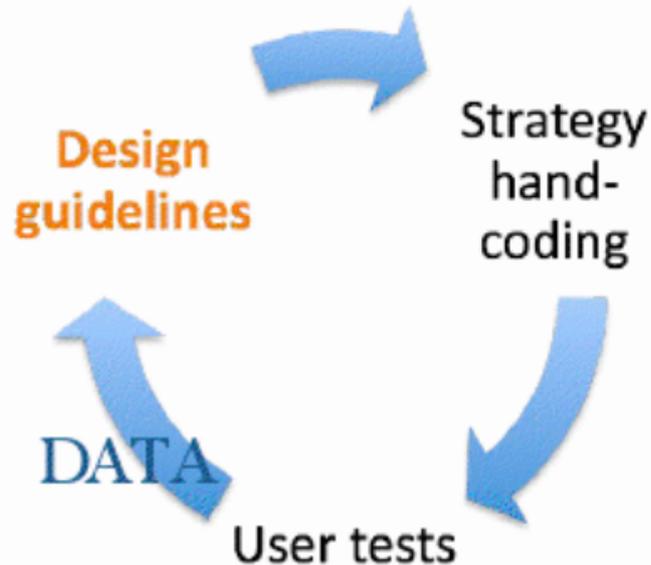
Allgemeines Vorgehen zum prob. DM:

- automatische Optimierung der Policy (maschinelles Lernen)
- basierend auf Interaktion mit echten oder simulierten Nutzern
- mittels reinforcement learning (ein Verfahren für Sequenzlernprobleme)

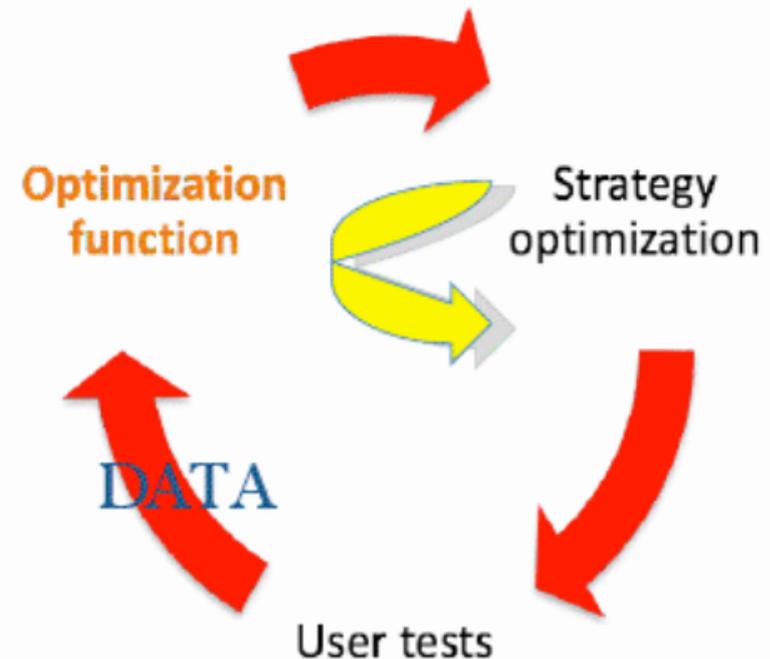
- am Anfang: “dumme” Dialog-Policy,
- Interaktion mit Nutzern, darauf basierend positives/negatives Feedback (“rewards”)
- Identifizierung von Fehlern in der Policy (warum ging es schief?)
- Korrektur der Fehler in der Policy
- repeat

Vergleich der Strategien

Conventional software life cycle



Automatic strategy optimisation



Design by "Best practices"

(Paek 2007)

Automatic design by optimization function

(= "programming by reward")

Vorgehen

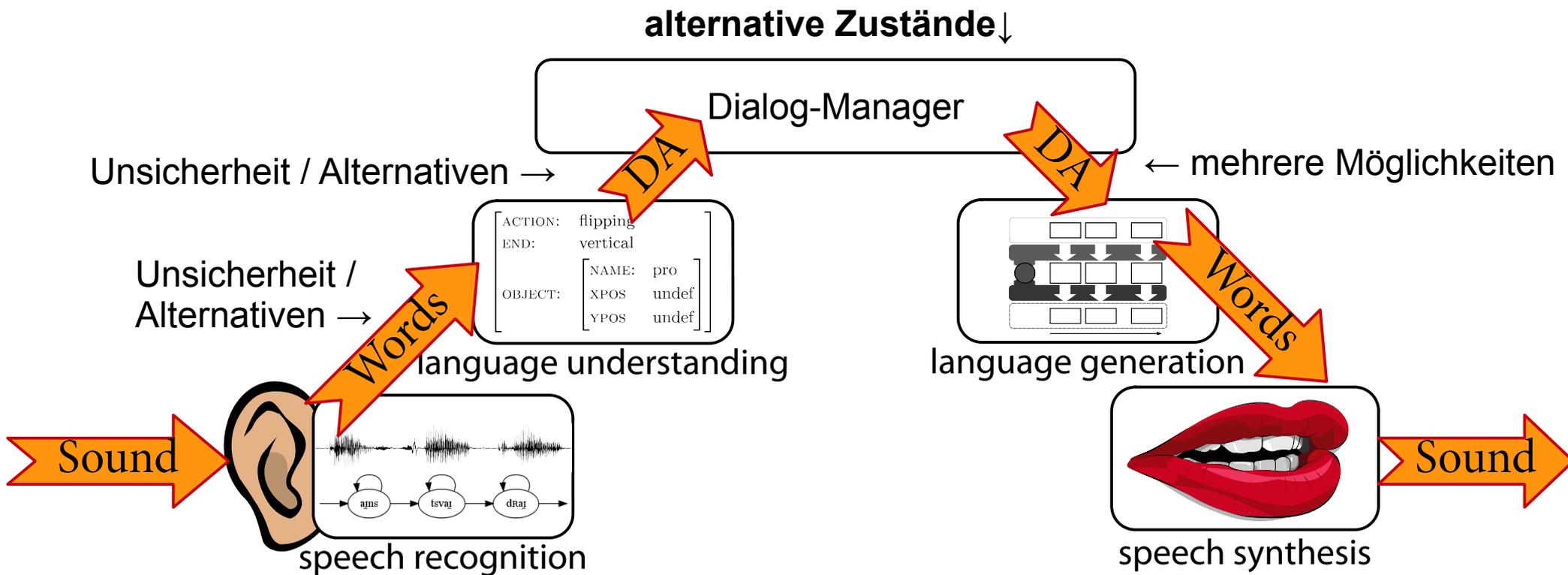
- *Planungsproblem* charakterisiert durch:
 - Zustandsraum (alle möglichen Dialogzustände)
 - Aktionsraum (alle möglichen Systemäußerungen)
 - Ziele (kodiert durch Rewards)
- Markov-Decision-Process
 - Zustand s aus S
 - Aktion a aus A
 - policy: $p: S \rightarrow A$

probabilistisch, also gewichtete Zufallsauswahl;
dann kann man die Gewichte lernen

Auswahl im Aktionsraum

- zunächst ein Klassifikationsproblem
 - Dialogzustand enthält notwendige Information
 - mögliche Systemantworten sind bekannt
 - Schwierigkeit vollständige Trainingsdaten zu bekommen
 - häufig gibt es mehrere mögliche nächste Aktionen
- sich widerstrebende Ziele:
 - möglichst sicher das richtige Flugticket buchen
= den Nutzer sicherlich richtig verstanden haben
 - möglichst schneller und einfacher Dialog ohne unendlich viele Rückfragen
 - Reward = (korrekte Buchung ? +10 : -30) – Länge des Dialogs
 - man weiß erst am Ende ob das Ziel erreicht ist

Ein noch probabilistischerer Dialogagent



Ziel: "beste" Möglichkeit auswählen

POMDP-basiertes Dialogmanagement

- Dialogmanager schätzt eine Verteilung über mögliche Dialogzustände

*was könnte wohl gemeint sein
und wie wahrscheinlich wäre das jeweils?*

- schätzt (für jeden möglichen Zustand) die Qualität über alle möglichen wählbaren Aktionen (die sich aus den Zustandsmöglichkeiten ergeben)
- wählt die Aktion die insgesamt am besten ist, weil:
 - sie wahrscheinlich richtig ist
 - sie hilft möglichst schnell Unsicherheit zu reduzieren
 - sie das Unwissen des Systems transparent macht
 - ...

Herausforderungen

- Trainingsdaten
 - bzw. “Trainer”, der auf Systemfehler eingehen muss
 - man kann nicht einfach aus fertigen Dialogen lernen
 - häufig Rückgriff auf “simulierte Nutzer”
(=einfache, regelbasierte Systeme)

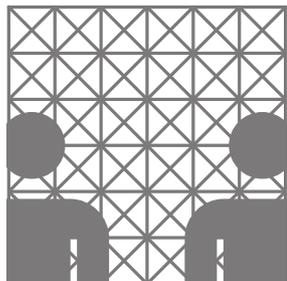
Vielen Dank.

baumann@informatik.uni-hamburg.de



<https://nats-www.informatik.uni-hamburg.de/SDS20>

Universität Hamburg, Department of Informatics
Language Technology Group



Notizen

Further Reading

- Sprachverstehen:
 - Gokhan Tur & Renato De Mori: Spoken Language Understanding, insbesondere Kapitel 3.
 - Jurafsky & Martin, Chapter 26, Sections 3 und 4

Desired Learning Outcomes

- Die Studierenden wissen, wie ein Dialogmodell auf Basis von Frames und Frame-Semantik eine Mischung aus System- und Nutzerinitiative ermöglicht, und können entsprechende Dialogmodelle entwerfen
- Die Studierenden verstehen, wie unsicheres Wissen in ein probabilistisches Dialogmodell integriert werden kann
 - Sie kennen den qualitativen Unterschied, ob das System einen Zustand oder eine Verteilung über Zustände nutzt.