Dialogue Engineering

Dialogue Systems

Amandine Decker, *amandine.decker@loria.fr*

M2 NLP 2025–2026

Recap'

- Dialogue is complex and can be characterised through many features...
  - These features vary depending on the type of conversation and their properties influence the flow of dialogue;
  - Yet dialogue comes quite easily to us.
- Designing dialogue systems requires to...
  - Formalise the structural backbone of a conversation (i.e. steps/phases, topical structure, etc.);
  - While balancing interactional features to produce effective dialogues (e.g. grounding, initiative, etc.).

Today

- Human Conversational Features in Dialogue Systems :
  - Overview of three dialogue features that must be operationalised when building a DS ;
  - Virtually all the features we discussed last time and probably others should be carefully addressed ;
- Dialogue Management & State Tracking :
  - Two modules of a DS pipeline ;
  - Crucial in task-oriented DS design ;
- Approaches to Building a Dialogue System :
  - Rule-based vs. neural-based ;
  - Several levels of complexity ;
- Ethical Issues in Dialogue System Design.

Human Conversational Features in Dialogue Systems

Grounding in dialogue systems I

- Grounding is the process of making sure that dialogue participants perceive, understand and accept each other's utterances.
- This includes :
  - Providing feedback to user utterances (e.g. ask user to confirm filled slots);
  - Dealing with responses to system feedback to user utterances (e.g. user indicating that system got it right or wrong);
  - Understanding user feedback to system utterances (e.g. "please repeat!");
  - Reacting appropriately to user feedback to system utterances (e.g. repeat utterance).

Grounding in dialogue systems II

- Often, grounding in dialogue system is reduced to one of the following :
  - Explicit confirmation

    U : I want to know timetables from Madrid.
    S : Do you want to leave from Madrid ?
    U : Yes.
  - Implicit confirmation

    U : I want to know timetables from Madrid.
    S : What time do you want to leave from Madrid ?
  - The latter is more efficient, but risky

    U : No, I just wanted to know about times from Madrid but I might be departing from somewhere else depending on whether I have the use of the car next Friday.

Grounding in dialogue systems III

- Confirmation only addresses a small fraction of human grounding behaviours :
  - Backchannels, clarification requests, negative feedback, rejection, ...
- In recent years, many systems try to avoid confirmation altogether, since it may be perceived as repetitive and annoying :
  - This is also risky, since the system may act on misheard or misunderstood information ;
  - The seriousness of this risk depends on the application in question (e.g. medical diagnosis vs. social chit-chat).

Grounding – Summary

- Grounding ensures understanding and acceptance.
- Approaches to grounding :
  - Explicit confirmation ;
  - Implicit confirmation.
- Risks of skipping confirmation :
  - Misunderstanding can lead to errors ;
  - Importance varies with application (e.g. medical vs. casual).

Turn taking in dialogue systems I

- Humans are good at turn-taking :
  - Very few overlaps in human conversation ;
  - Easy repairs when overlaps occur ;
  - Very short pauses between utterances.
- In contrast, many systems are not very flexible with respect to turn-taking.
- Strict turn-taking :
  - User cannot speak while system speaks ;
  - When user speaks, a pause of T leads to system taking the turn.
- Strict turn-taking can cause the system to interrupt the user, but the system will keep speaking even if the user tries to take the turn.
- These problems can be addressed by barge-in and end-of-turn detection.

Turn taking in dialogue systems II

- Barge-in :
  - The system listens for user speech while it's speaking;
  - This allows users to take the turn from the system;
  - A variant is "multimodal barge-in", where the user can push a button to take the turn.
- End-of-turn detection :
  - The system combines several factors such as pauses, intonation, syntactic completeness and semantic completeness to decide when user turn is done;
  - This helps avoid system interrupting the user.

Turn-Taking – Summary

- Challenges compared to human turn-taking :
  - Limited flexibility.
  - Risk of interruptions.
- Solutions :
  - Barge-in : User interrupts system.
  - End-of-turn detection : Based on pauses, intonation, etc.

Initiative in dialogue

- The dialogue participant who has initiative...
  - Initiates interactions (task initiative);
  - Produces first part of adjacency pairs (dialogue initiative) :

    ask questions;
    make requests;
    make assertions;
  - Decides the topic of conversation and when to change topic (dialogue/task initiative).
- In everyday human-human dialogue, initiative shifts back and forth between participants.
- Many dialogue systems are more limited w.r.t. initiative.

User initiative dialogue systems

- The user has initiative and directs the system.
- The user asks a question (or makes a request) :
  - USR : “What is the weather tomorrow in Gothenburg?”
  - SYS : “Rain.”
- The system answers (or carries out action) :
  - USR : “Play some jazz”
  - SYS : [plays some jazz]
- The system does not ask questions to the user or engage in clarification or confirmation.

  $\rightarrow$ Used for simple database queries or command-style interaction.

System initiative dialogue systems

- The system has initiative and directs the user.
- Example :
  - SYS : Do you want to make a payment or get your account statement ?
  - USR : Make a payment
  - SYS : Please say your credit card number
  - USR : NNNN NNNN NNNN
  - SYS : Please say the account number
  - USR : XXX XXX XXX
  - SYS : Please say the amount
  - USR : YYY
  - SYS : Transfer complete.
- Example : If user does anything other than what the system asks, the system will not hear or understand :
  - SYS : Please say the amount.
  - USR : How much do I have on my account now ?
  - SYS : I did not catch that. Please say the amount.

Mixed initiative dialogue systems I

- Conversational initiative can shift between system and user.
- We will look at mixed initiative in form-filling dialogue here, but it is also relevant in other kinds of dialogue.
- In form-filling dialogue, the structure of the form governs the dialogue.
- Example form : make a call
  - Person to call : ___
  - Number type (Mobile or Work) : ___

Mixed initiative dialogue systems II

- A minimal degree of mixed initiative is to allow the user to freely state their question or request, rather than selecting from a limited list :
  - USR : Hello Virtual Assistant!
  - SYS : [shows “open mic” icon on screen]
  - USR : I want to make a call
  - SYS : Who do you want to call ?
  - USR : Kim
  - SYS : Mobile or work number ?
  - USR : Mobile.
  - SYS : Calling Kim’s mobile.

Mixed initiative dialogue systems II

- A minimal degree of mixed initiative is to allow the user to freely state their question or request, rather than selecting from a limited list :
  - USR : Hello Virtual Assistant!
  - SYS : [shows “open mic” icon on screen]
  - USR : I want to make a call
  - SYS : Who do you want to call ?
  - USR : Kim
  - SYS : Mobile or work number ?
  - USR : Mobile.
  - SYS : Calling Kim’s mobile.
- The system then inquires for the necessary information.

Mixed initiative dialogue systems II

- A minimal degree of mixed initiative is to allow the user to freely state their question or request, rather than selecting from a limited list :
  - USR : Hello Virtual Assistant!
  - SYS : [shows “open mic” icon on screen]
  - USR : I want to make a call
  - SYS : Who do you want to call ?
  - USR : Kim
  - SYS : Mobile or work number ?
  - USR : Mobile.
  - SYS : Calling Kim’s mobile.
- The system then inquires for the necessary information.
- The above is still not very flexible.

Mixed initiative dialogue systems III

- A slightly higher degree of flexibility and mixed initiative is achieved by allowing over-answering : allowing the user to provide more (or other) information than requested :
  - USR : I want to make a call
  - SYS : **Who** do you want to call ?
  - USR : **Kim**'s mobile
  - SYS : Calling Kim's mobile.

Mixed initiative dialogue systems IV

- A variant of this is to also allow for one-shot requests (similar to user-initiative dialogue) :
  - USR : Call Kim's mobile
  - SYS : Calling Kim's mobile.
- A further variant, ask for missing information :
  - USR : Call Kim
  - SYS : Mobile or work number ?
  - USR : Mobile.
  - SYS : Calling Kim's mobile.

Mixed initiative dialogue systems V

- Yet more mixed initiative is allowed if users can ask questions freely (user initiated subdialogues) :
  - USR : Call Kim
  - SYS : Mobile or work number ?
  - USR : Hmmm... What time is it ?
  - SYS : 6pm.
  - USR : OK, so mobile.
  - SYS : Calling Kim's mobile.

Mixed initiative dialogue systems VI

- The more of these dialogue features a system can handle (including turn-taking and grounding), the more mixed initiative it offers.
- So mixed initiative is a matter of degree.
- Alternatively, we can stop talking about initiative as a global feature, and talk about specific features instead (over-answering, user initiated subdialogues, etc.).

Initiative – Summary

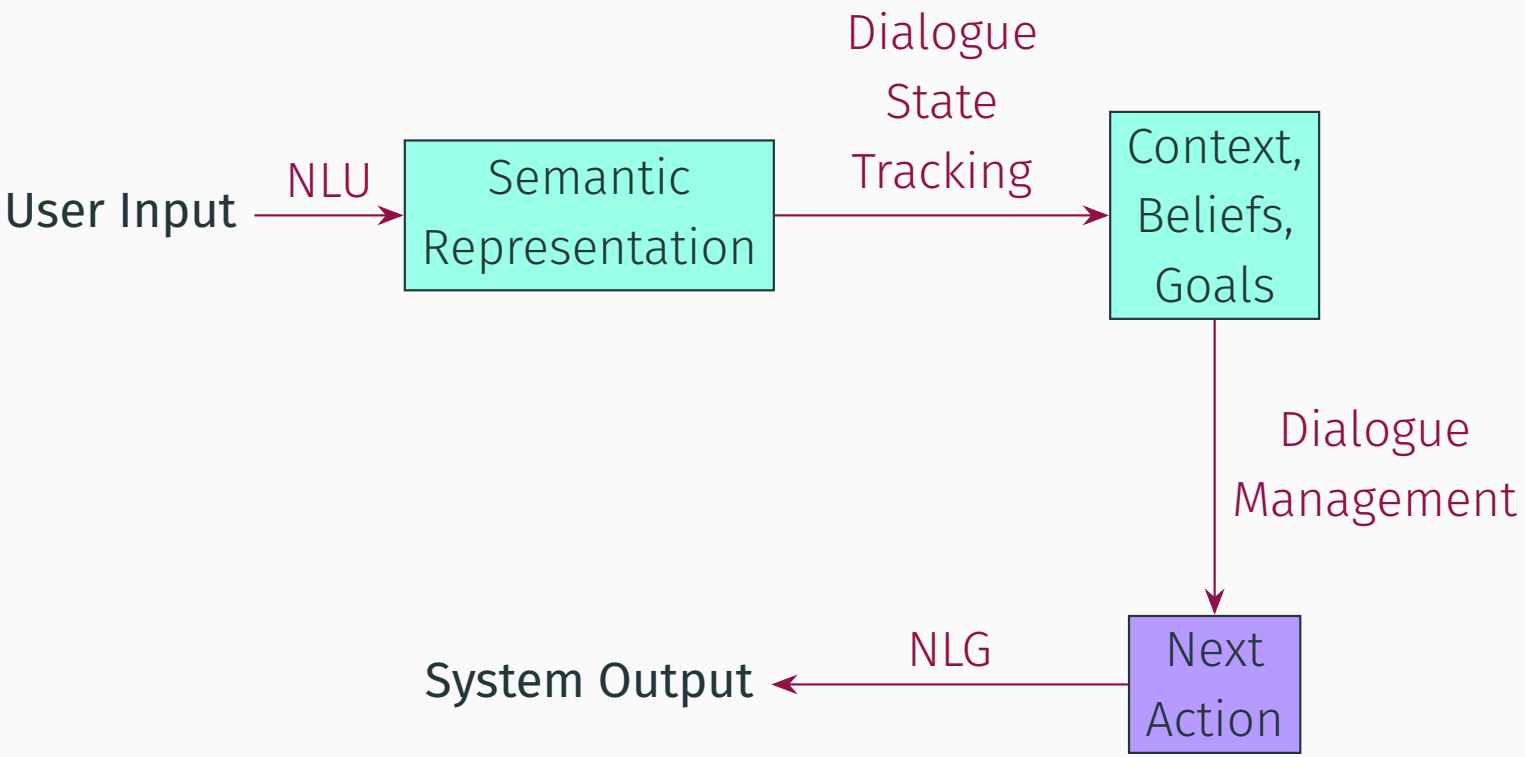
- Initiative defines who drives the conversation.
- Types :
  - User initiative : User asks, system responds.
  - System initiative : System leads, user follows.
  - Mixed initiative : Flexible shifting of roles.

Conversational Features in DS

- Grounding, turn-taking, and initiative must be explicitly modelled in the design of a DS :
  - The approach changes the training data, the technical functioning, the potential applications, etc.
- Other features must be discussed even though many tend to be assumed and/or overlooked :
  - DSs are primarily meant for one user at a time;
  - They are often in written form;
  - The physical environment is not accessible;
  - etc.
- Clearly defining the behaviour of conversational features enables to manage expectations and design a system adapted to a given need.

Dialogue Management & State Tracking

Typical Dialogue System Pipeline



Two classes of systems

- Goal-based (or task oriented) dialogue agents / systems :
  - Important that the systems are reliable and controllable ;
  - Rogue behaviour (e.g. biased or harmful output) not an option.
- Chatbots – originally mostly for fun (but users come up with practical uses) :
  - Focus on open domain interaction ;
  - Entertainment value more important than reliability and control (ethical questions).

Dialogue State Tracking (DST) I

- Tracks what the system believes is true about :
  - User goals (intents);
  - Slot values / constraints;
  - Dialogue history;
  - Uncertainty (ASR, ambiguous phrasing).
- Acts as the “memory” of the conversation.

Dialogue state tracking (DST) II

- A dialogue state tracker observes signals from ASR and NLU components and accesses content in external databases or knowledge bases.
  - State = assignment of values to slots in the system (so strictly form-based);
  - Input : a set of possible dialogue state hypotheses;
  - Output : probability distribution over the set of hypotheses, a.k.a. belief state.

Sources of Uncertainty

- Speech recognition errors;
- Implicit / underspecified user requests;
- Conflicting or changed user goals;
- Noisy environments.

DST must continuously update beliefs about the state.

Task-oriented systems and reinforcement learning

- Most task-oriented dialogue systems use slot-filling methods to capture user intent in a domain specific conversation.
  - A task needs to be pre-defined by a set of manually crafted states with multiple slots;
  - Susceptible to uncertainty :

    USR : Book me a flight to Gothenburg on September 3.
    USR : No wait, September 2.
- Recent task-oriented systems cast dialogue as a reinforcement learning problem where rewards are given based on how well a task is completed;
- They are recently boosted by deep reinforcement learning, which eliminates the need for feature engineering.

Neural Dialogue State Tracking

- Neural encoders encode user + system utterances;
- Predict slot-values directly (classification or span extraction).
- Typically based on Transformers.
- State-of-the-art on MultiWOZ and related benchmarks.

DST Approaches – Summary

- **Rule-based trackers**
  - Deterministic updates;
  - Transparent but brittle.
- **Probabilistic Trackers**
  - Model uncertainty explicitly;
  - Loss of explainability.
- **Hybrid models**
  - Rules + probabilistic updates.

In all cases some base data is needed!

What is Dialogue Management ?

- The component that decides :
  - **What the system should do next** ;
  - How to respond to the current context ;
  - How to manage task progress and errors.
- Takes dialogue state as input and outputs an **action**.

Why is dialogue management important?

- Without a DM, there is no dialogue.
- The user has to give all information that the system needs in a single utterance, which in some cases may be very difficult and cognitively demanding :
  - “I want to book a flight from Gothenburg to London on September 2 in the afternoon, coming back on the 10th in the morning, for 2 adults and 2 children aged 5 and 8, with no stopovers and preferably going to Heathrow airport, economy class.”
- If any information is left out, there is no way to supply it later.

Why is dialogue management important?

- A dialogue manager makes it possible to have coherent exchanges consisting of several turns;
- This means that the user does not have to say everything at once;
- Instead, the user can say what's on her mind, and the system will ask for additional needed information.

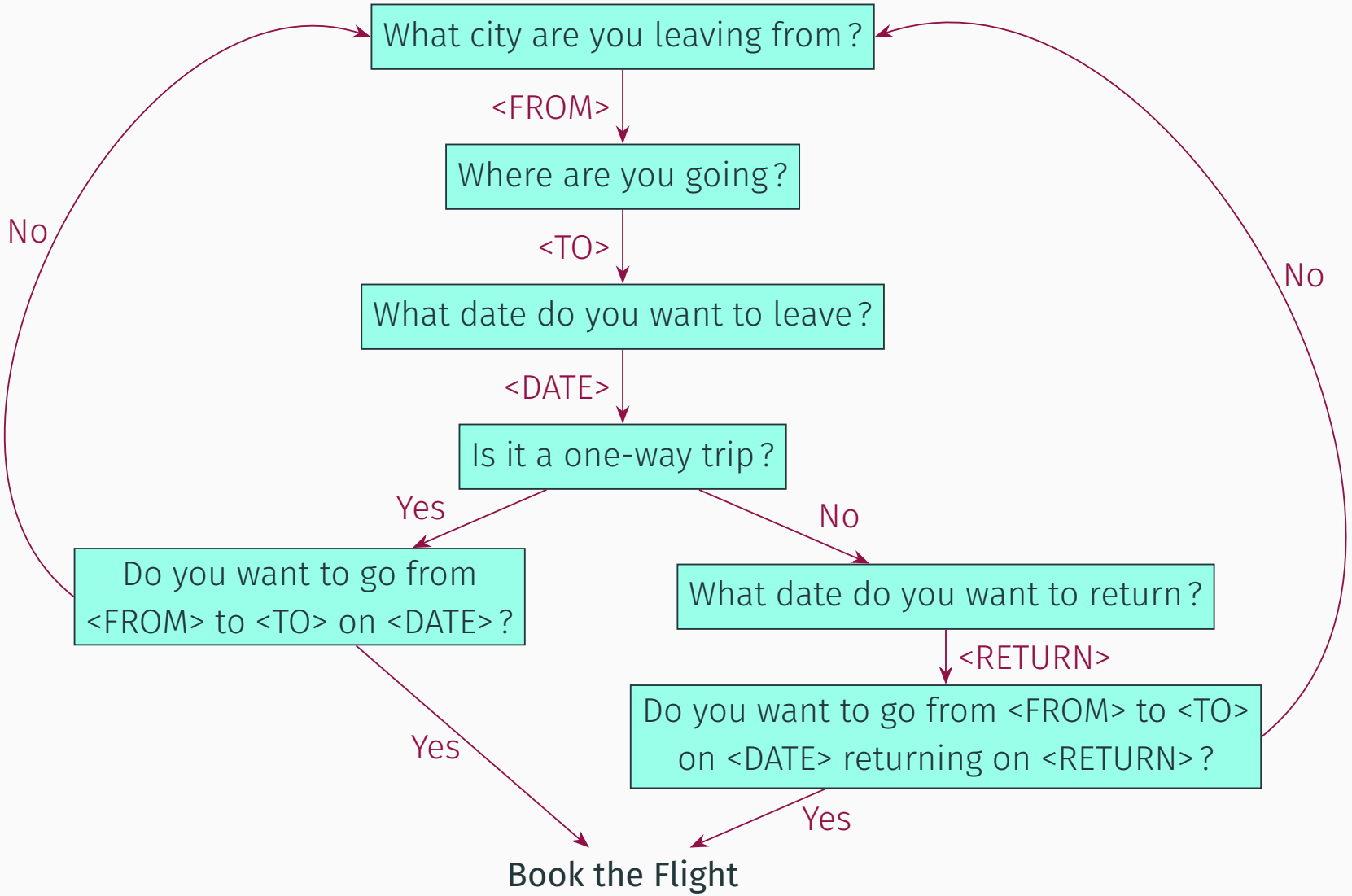
Dialogue Manager responsibilities

- Control conversation flow;
- Maintain coherence and handle grounding;
- Select clarification strategies;
- Guide system or user initiative;
- Coordinate modules (NLU, DB, NLG).

Dialogue Management methods

- Rule based :
  - Finite-state;
  - Form-based;
  - Plan-based;
  - Information state update.
- Statistical data-driven (machine learning);
- Neural end-to-end.
- Task oriented dialogue systems tend to rely on rule-based DM; reliable and predictable but less flexible.
- Chatbots are often based on statistical or neural end-to-end (ChatGPT) approaches; more flexible but less reliable.

Finite state-based DM I



Finite state-based DM II

- Represents dialogue flow using a finite state machine :
  - States : questions to the user ;
  - Transitions : user responses and resulting actions ;
  - Also stores answers in variables (<DATE> etc.) (not pure finite state).
- Works for system initiative dialogue :
  - System has all the initiative ;
  - Tends to ignores or misinterpret anything which is not a direct answer to a system question.

Finite state-based DM III

- However, human-human conversation is very often mixed initiative :
  - User may provide unrequested information ;
  - User may ask a question in response to a question ;
  - etc.
- This requires to create and maintain many more states and transitions.

Form-based DM I

- Form = slots and values.
- Relies on the structure of a form to guide the dialogue.
- Provides some aspects of mixed initiative dialogue.
- Asks the user questions to fill slots in the frame...
  - but allows the user to guide the dialogue by giving information that fills other slots in the frame.
- Each slot may be associated with a question to ask the user, following type :
  - ORIGIN CITY “From what city are you leaving ?” ;
  - DESTINATION CITY “Where are you going ?” ;
  - DEPARTURE TIME “When would you like to leave ?” ;
  - ARRIVAL TIME “When do you want to arrive ?”.

Form-based DM II

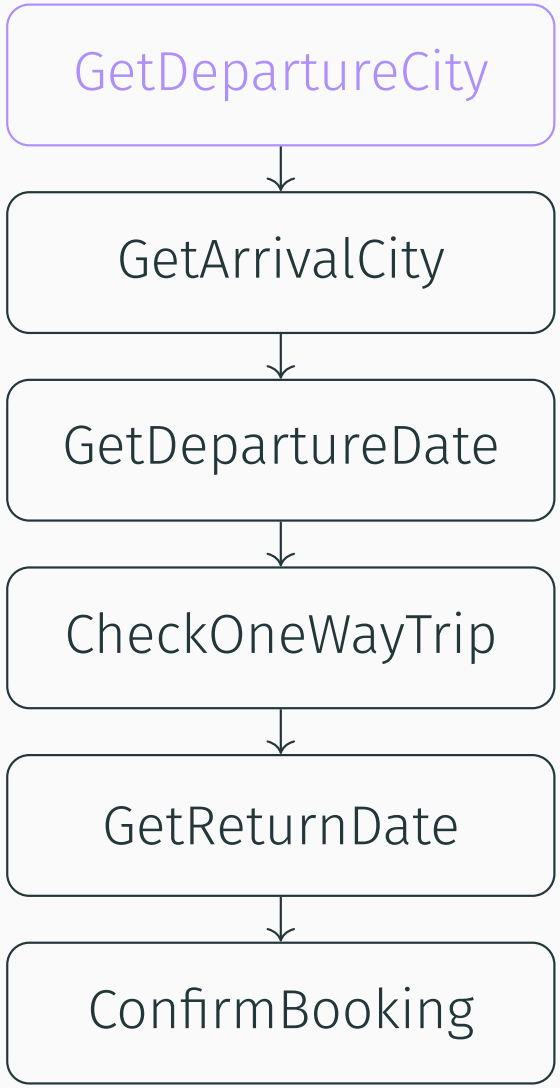
- DM asks questions to the user, filling any slot that the user specifies...
  - ...until it has enough information to perform a data base query, and then return the result to the user.
- If the user happens to answer two or three questions at a time, the system has to fill in these slots and then remember not to ask the user the associated questions for the slots.
- Does away with the strict constraints that the finite-state manager imposes.

Plan-based DM I

- Popular 1980's-1990's;
- View dialogue as planning and plan-recognition;
- Highly general approach, can handle very complex dialogues (in principle).
- However :
  - Adapting such approaches to individual domains is very labour-intensive;
  - Systems are very brittle and tend to break easily.

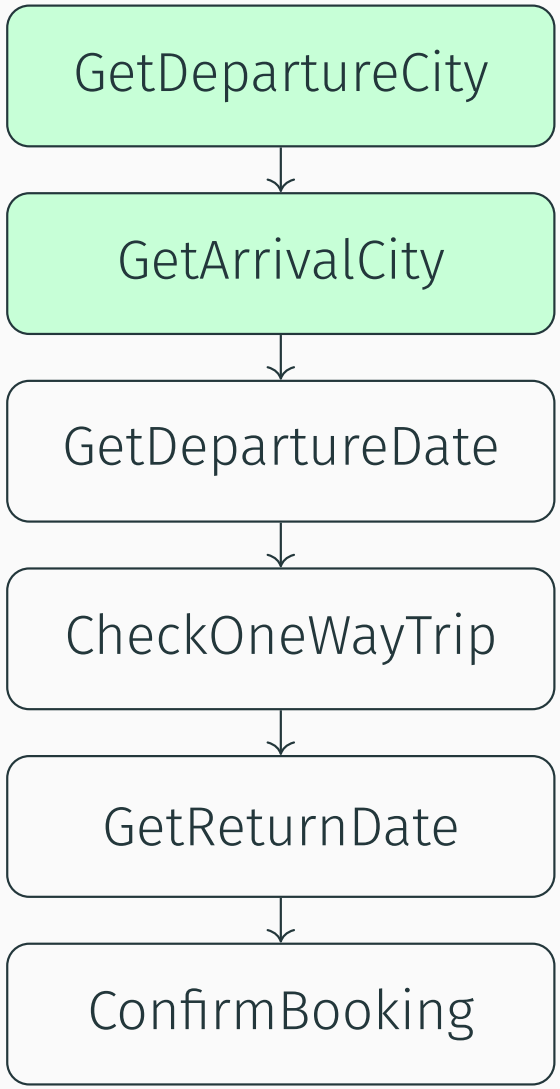
Plan-based DM II

System : Hi ! Where will you be flying from ?



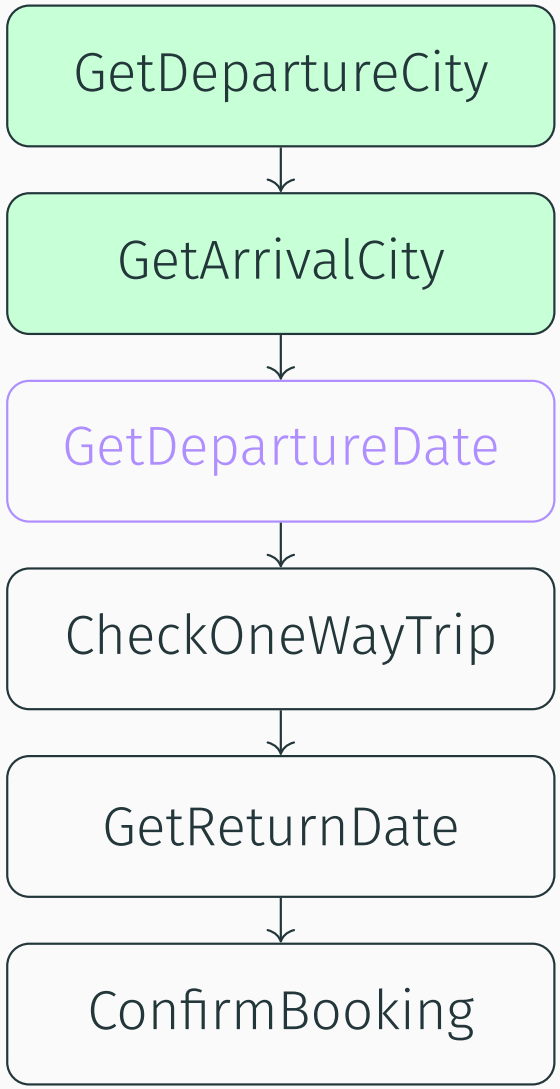
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.



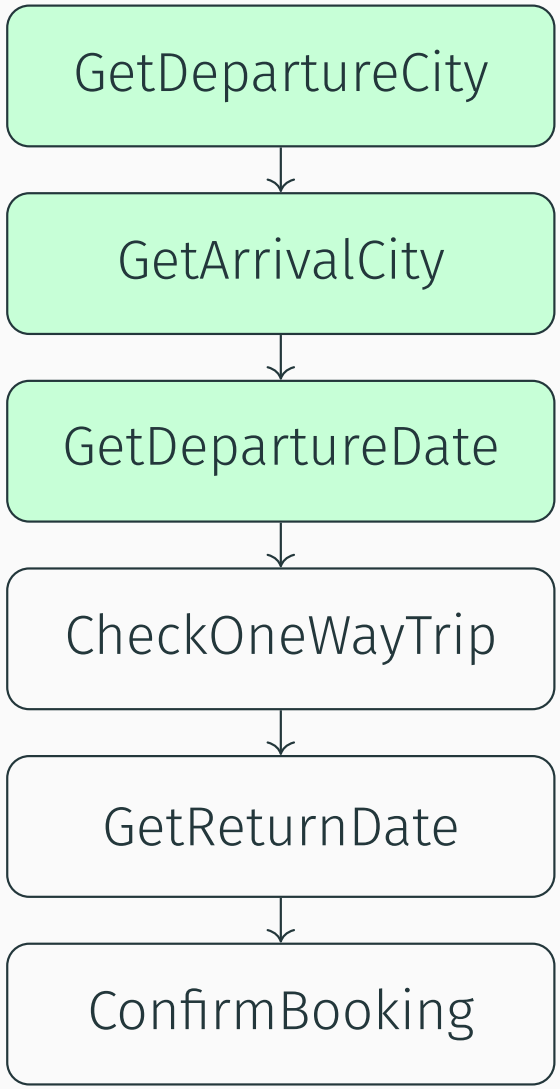
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?



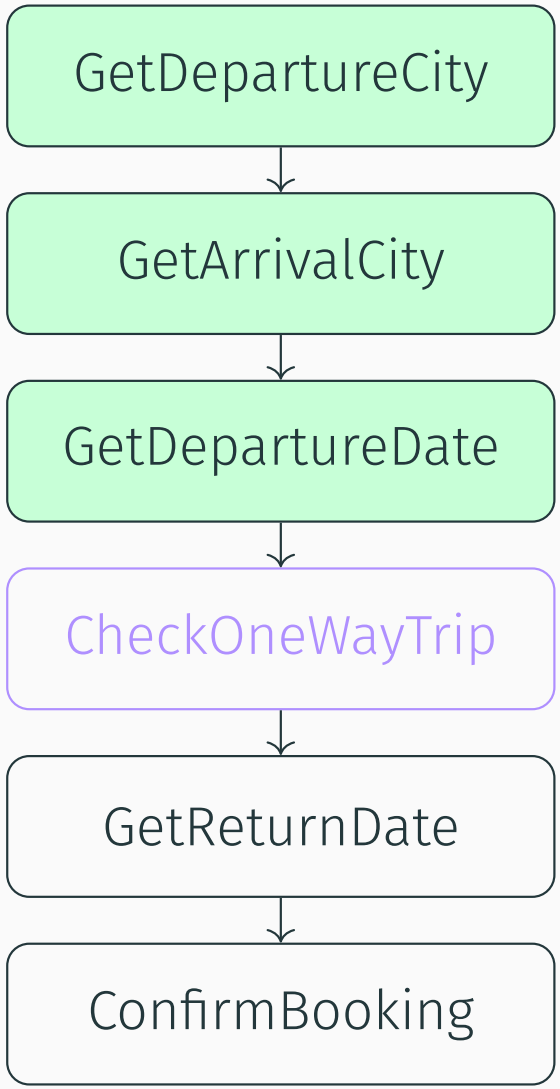
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.



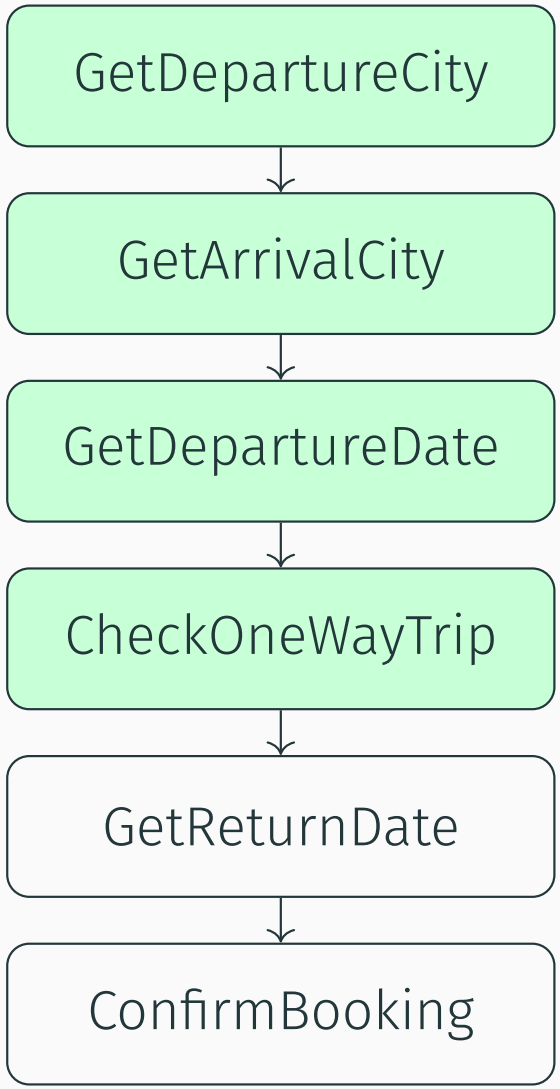
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?



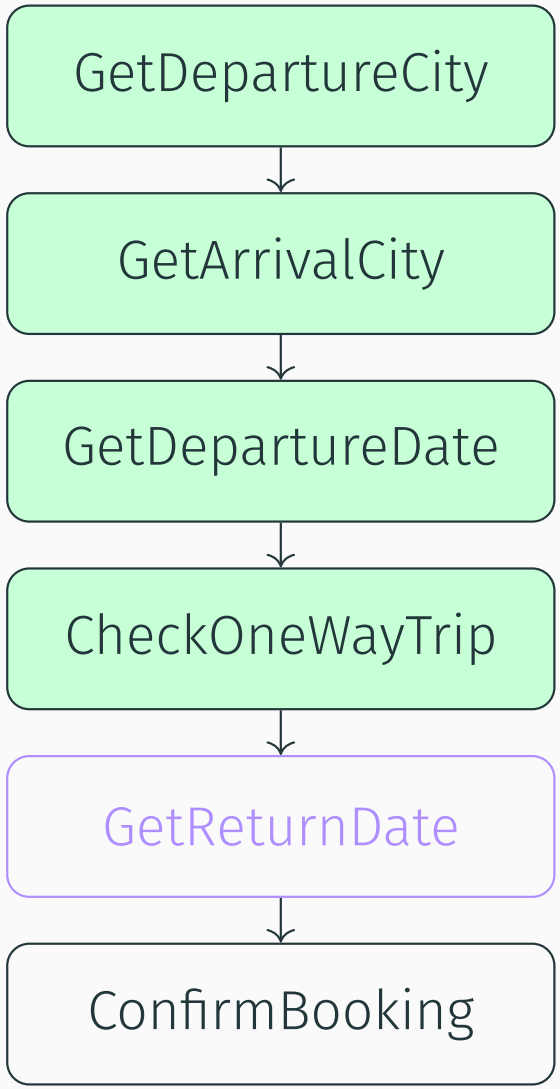
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?
User : No, it's not.



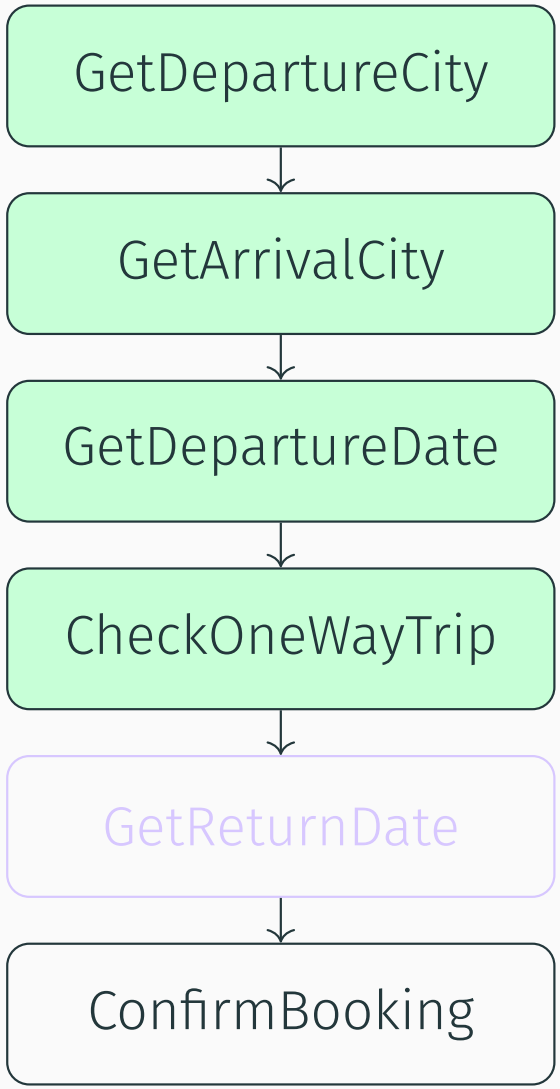
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?
User : No, it's not.
System : When would you like to return ?



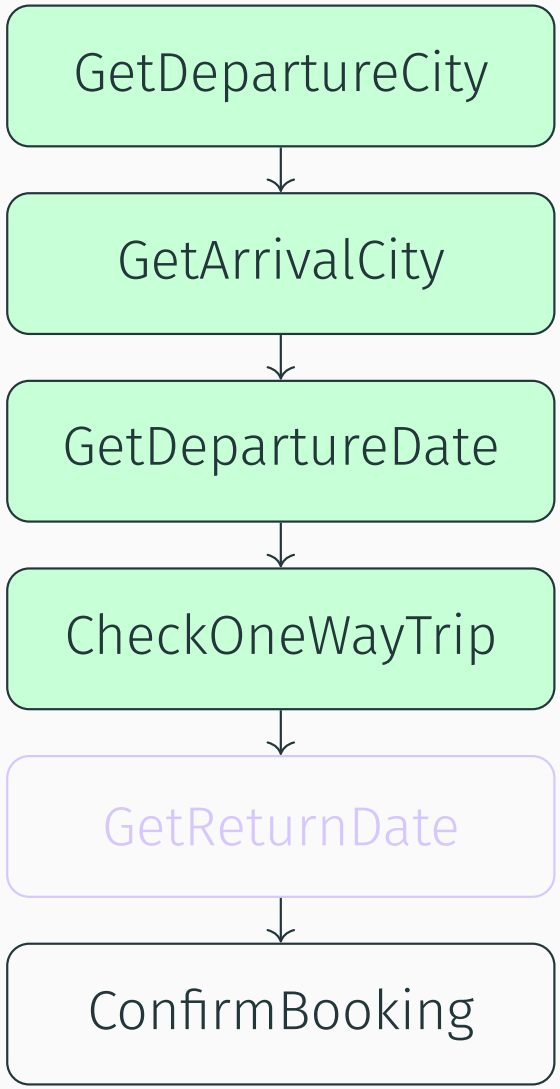
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?
User : No, it's not.
System : When would you like to return ?
User : Uh, do I have anything planned the week of December 15th ?



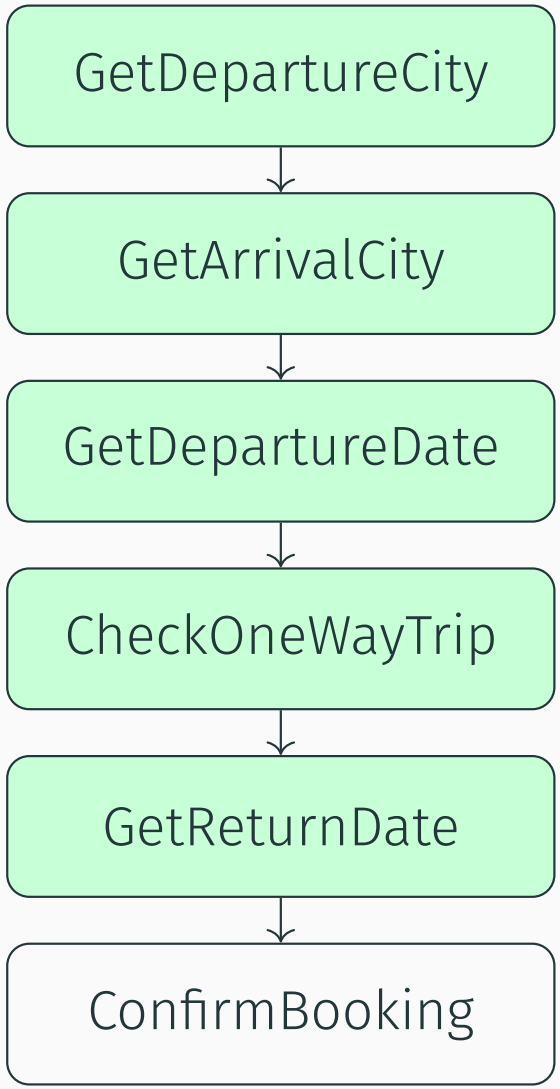
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?
User : No, it's not.
System : When would you like to return ?
User : Uh, do I have anything planned the week of December 15th ?
System : You have a presentation at CLASP on December 16th, but nothing else.



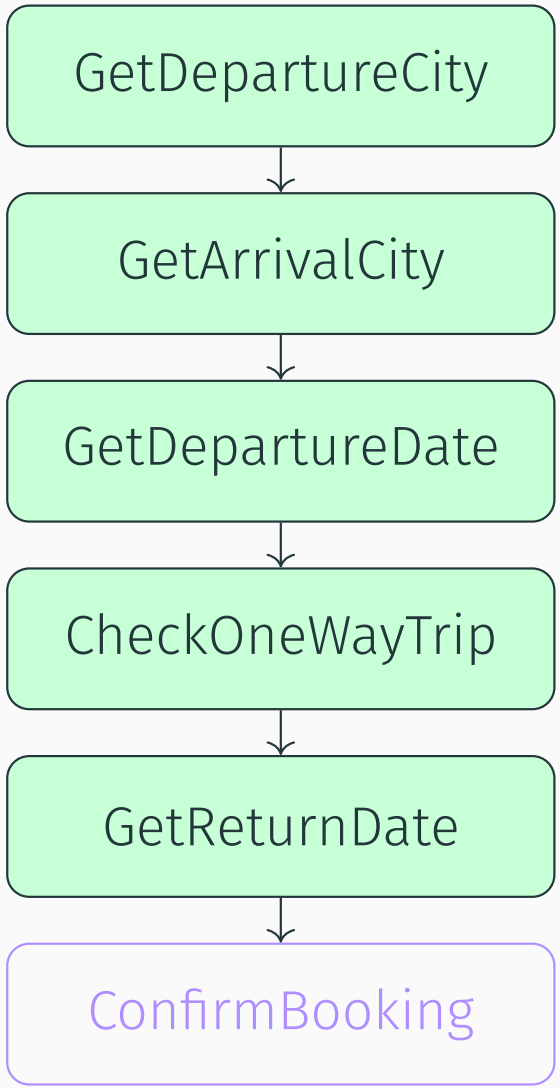
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?
User : No, it's not.
System : When would you like to return ?
User : Uh, do I have anything planned the week of December 15th ?
System : You have a presentation at CLASP on December 16th, but nothing else.
User : Okay, then I'd like to return on December 19th.



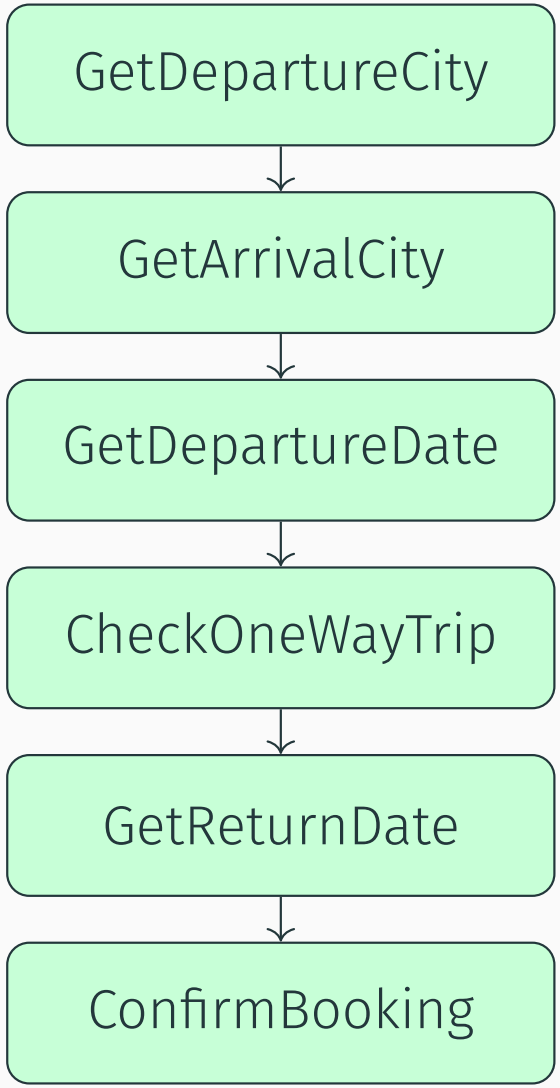
Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?
User : No, it's not.
System : When would you like to return ?
User : Uh, do I have anything planned the week of December 15th ?
System : You have a presentation at CLASP on December 16th, but nothing else.
User : Okay, then I'd like to return on December 19th.
System : Let me confirm : Luxemburg → Gothenburg, departing Dec 14, returning Dec 19. Correct ?



Plan-based DM II

System : Hi ! Where will you be flying from ?
User : I'm flying from Luxemburg to Gothenburg.
System : Great ! What's your departure date ?
User : December 14th.
System : Is this a one-way trip ?
User : No, it's not.
System : When would you like to return ?
User : Uh, do I have anything planned the week of December 15th ?
System : You have a presentation at CLASP on December 16th, but nothing else.
User : Okay, then I'd like to return on December 19th.
System : Let me confirm : Luxemburg → Gothenburg, departing Dec 14, returning Dec 19. Correct ?
User : Yes.



Information State approach

- Goal : explore the space between finite-state/form-filling approaches (robust but limited) and plan-based approaches (capable but brittle and labour-intensive).
- Key component : a rich Information State, representing the state of the dialogue so far.
- Deal with dialogue beyond form-filling in a robust way :
  - General and versatile approaches to confirmation, turn-management and other basic dialogue phenomena;
  - Dealing with multiple forms;
  - Comparing alternatives (“negotiative dialogue”);
  - Instructional dialogue (e.g. technical manuals);
  - Educational dialogue (e.g. homework exercises);
  - Problem-solving dialogue (e.g. putting together an itinerary).

Dialogue Management Approaches – Summary

- **Finite-State** :
  - Simple, predictable;
  - Good for fixed procedures.
- **Form** :
  - Flexible ordering;
  - Common in task-oriented domains (booking, queries).
- **Plan** :
  - Stack-like representation of tasks;
  - More scalable and adaptive.

DM + DST in Practice

- Modern systems :
  - May keep explicit state (API calls, structured slots);
  - Or rely on implicit state within large models.
- Trade-off :
  - **Explicit state** = control, reliability;
  - **Implicit state (LLMs)** = flexibility, naturalness.

Key Takeaways

- Dialogue State Tracking = maintaining beliefs about user goals;
- Dialogue Management = planning the system's next action;
- Approaches range from handcrafted to fully neural.

Approaches to Building a Dialogue System

Design guidelines I

- In rule-based systems conversation flow and other aspects of the interface are handcrafted using best practice guidelines :
  - Developed by voice user interface designers (Pearl, 2016 ; Batish, 2018).
- Guidelines on elements of conversations :
  - how to design effective prompts ;
  - how to sound natural ;
  - how to act in a cooperative manner ;
  - how to offer help at any time ;
  - how to prevent errors ;
  - how to recover from errors when they occur.

Design guidelines II

- Higher-level guidelines :
  - how to promote engagement and retention;
  - how to make the customer experience more personal and more pleasant;
  - the use of personas and branding.
- Guidelines addressing linguistic aspects of conversational interaction :
  - maintaining the context in multi-turn conversations;
  - asking follow-up questions;
  - maintaining and changing topics;
  - error recovery.

Design guidelines III

- Guidelines concerned with social competence :
  - promoting engagement;
  - displaying personality;
  - expressing and interpreting emotion.
- Guidelines regarding psychological aspects :
  - being able to recognise the beliefs and intentions of the other conversational participant (= theory of mind).
- All of these aspects are important for a conversational agent to be effective and engaging.

User centered Design

- Study the user and task;
- Build simulations and prototypes;
- Iteratively test the design on users.

Study the user and task

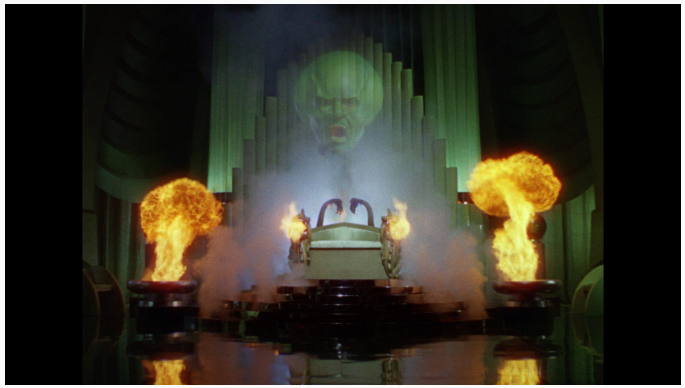
Understand the potential users and the nature of the task by :

- interviews with users ;
- study of related human-human dialogues :

  $\rightarrow$ collect, transcribe if spoken, analyse ;
- investigation of similar systems.

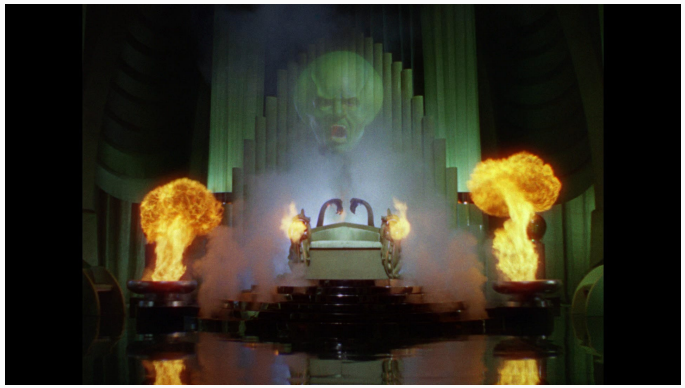
Build simulations I

- The users interact with what they think is a software agent



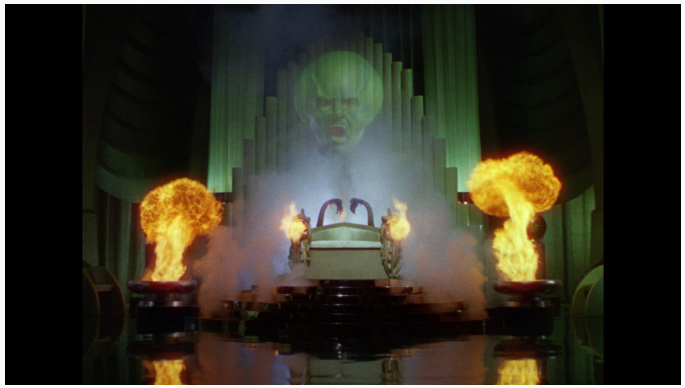
Build simulations I

- The users interact with what they think is a software agent but is in fact a human “wizard” disguised by a software interface.



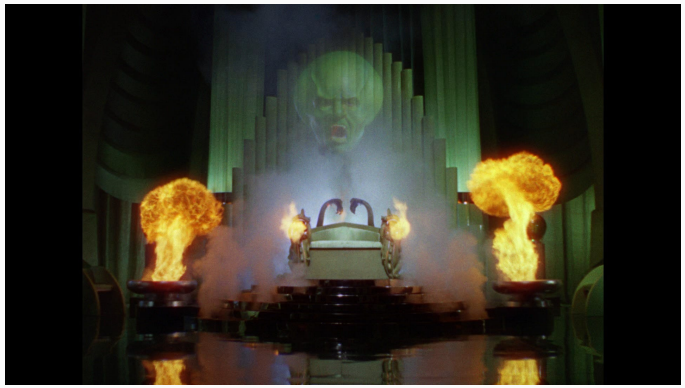
Build simulations I

- The users interact with what they think is a software agent but is in fact a human “wizard” disguised by a software interface.
- The Wizard of Oz (Baum, 1900) : the wizard turned out to be just a simulation controlled by a man behind a curtain or screen.



Build simulations I

- The users interact with what they think is a software agent but is in fact a human “wizard” disguised by a software interface.
- The Wizard of Oz (Baum, 1900) : the wizard turned out to be just a simulation controlled by a man behind a curtain or screen.
- Can be used to test out an architecture before implementation only the interface software and databases need to be in place.
- Results of wizard studies are thus somewhat idealised, but still can provide a useful first idea of the domain issues.



Build prototypes

- The alternative to doing WOz simulation is to build a scaled-down and incomplete version of the envisioned system.
- Testing with users may reveal unforeseen issues.
- Advantage over WOz is that the system may be more like the end product that the simulation is.
- Disadvantage is that more work may be needed to build the prototype, but that depends on whether it's an entirely new system (new DM, ASR, TTS, NLU, NLG, domain) or if it's an existing system which has been adapted to a new domain.

Iteratively test the design on users I

- An iterative design cycle with embedded user testing is essential in system design.
- The iterative method is also important for designing prompts that cause the user to respond in a way that the system can understand.
- User testing and evaluation is crucial in dialogue system design.

Iteratively test the design on users II

- Computing a user satisfaction rating can be done by having users interact with a dialogue system to perform a task, and then having them complete a questionnaire.
- It is often economically infeasible to run complete user satisfaction studies after every change in a system.
- For this reason it is often useful to have performance evaluation heuristics which correlate well with human satisfaction.

Types of evaluation

- Black-box evaluation : measure parameters related to
  - objective measures : task completion rate, word error rate, number of dialogue turns, time, etc.
  - subjective judgements : user satisfaction, perceived speed, understandability, predictability, etc.
- Glass-box evaluation :
  - look at the design of the system ;
  - the algorithms that are implemented ;
  - the linguistic resources it uses (e.g. vocabulary size), etc.
  - Often difficult to predict performance only on the basis of glass-box evaluation
  - However, more informative with respect to error analysis or future developments of a system.

Ethical Issues in Dialogue System Design

Replicating biases in training data I

- Machine learning systems replicate and **reinforce** biases that occurred in the training data.
- Bias in training data can arise from gaps in the training data, for example, a lack of diversity that excludes females, people of color, and people from different religious and cultural backgrounds.
- Bias can also be introduced unintentionally by annotators.
- Bias in Conversational AI is being addressed by the Conversation AI Research Github Organization.

Replicating biases in training data II

- Microsoft's Tay chatbot
  - Went live on Twitter in 2016;
  - Taken offline 16 hours later;
  - In that time it had started posting racial slurs, conspiracy theories, and personal attacks;
  - Learned from user interactions (Neff and Nagy 2016).
- Dialogue datasets
  - Henderson et al. (2018) examined standard datasets (Twitter, Reddit, movie dialogues);
  - Found examples of hate speech, offensive language, and bias (both in the original training data, and in the output of chatbots trained on the data).

Replicating biases in training data III

- Preventing offensive output was a major concern in the Alexa Prize :
  - Bots were trained from publicly available data sources such as Reddit and Twitter that contain large amounts of sensitive content.
- ChatGPT FAQ quote : While we've made efforts to make the model refuse inappropriate requests, it will sometimes respond to harmful instructions or exhibit biased behaviour.

Recording sensitive data

- This was noticed already with Eliza in the 1960's.
- Agents may record sensitive data :
  - e.g. "Computer, turn on the lights [answers the phone] Hi, yes, my password is...".
- This recording may then be used to train a seq2seq conversational model.
- Henderson et al. (2018) showed they could recover such information by giving a seq2seq model keyphrases (e.g., "password is").

Dealing with offensive input II

- Curry and Rieser (2019) conducted a crowd-based evaluation of abuse response strategies in conversational systems finding that strategies such as “polite refusal” scored highly.
- Curry and Rieser (2018) examined the responses of commercial systems to inappropriate content such as bullying and sexual harassment.
- They collected and annotated a corpus of data (the #MeTooAlexa corpus) based on around 370,000 conversations from the Alexa Prize 2017 and from 11 state-of-the-art systems.

Dealing with sensitive input

- There are also inputs that are inappropriate or sensitive.
- Worswick (2018) discusses various types of sensitive messages sent to the chatbot Mitsuku (romantic attention, suicidal thoughts and serious issues, etc.);
- This is important in a context where a part of open chatbots' users consider them as close friend / romantic partner and pour real feelings in a completely imbalanced interaction.

Misleading users

- Dialogue systems may provide faulty information to users, output fake news, or give inaccurate or inappropriate advice.
- There is a need for explanation in AI to engender trust and transparency (explainable AI).
- ChatGPT FAQ quote :
  - ChatGPT sometimes writes plausible-sounding but incorrect or nonsensical answers. Fixing this issue is challenging, as : (1) during RL training, there's currently no source of truth ; (2) training the model to be more cautious causes it to decline questions that it can answer correctly ; and (3) supervised training misleads the model because the ideal answer depends on what the model knows, rather than what the human demonstrator knows.

Conversational AI for social good

- AI for social good aims to create safer and better-behaved conversational AI models.
- Wang et al. [2019] developed intelligent persuasive conversational agents :
  - Aim : change people's opinions and actions for social good ;
  - Based on research in persuasion in the social sciences.
- Given that persuasion can be used for evil as well as good causes, the authors raise various ethical concerns such as :
  - which scenarios are appropriate for the use of automated persuasion ;
  - the need to keep users informed of the role of the dialogue system ;
  - giving the users the option to communicate directly with the humans behind the system ;
  - developing procedures to monitor the responses generated by the system to ensure that they are appropriate and comply with ethical standards.

Summary

Summary

- There are two aspects to dialogue systems :
  - Their ability to tackle the task they were designed for ;
  - Their conversational capacities.
- Both aspects must be considered regardless of approach :
  - Rule-based, neural-based, or hybrid systems ;
  - Complexity varies : an open-domain, perfectly interactional chatbot is not always required.
- Ethical implications arise from the system's functioning, purpose, and usage.
- Before we reach the use stage, two more crucial aspects must be discussed.
  - **Resources :** corpora to build and train dialogue systems ;
  - **Evaluation :** methods to measure system performance.

Next Time

- Resources, i.e. corpora to build and train dialogue systems :
  - Basis of building guidelines, understanding how dialogue works, etc.
  - Used as gold standards for evaluation purposes ;
  - The type of resources used influences the systems we build.
- Evaluation, i.e. methods to measure system performance :
  - Again two aspects to assess : task and conversation ;
  - Ideally user feedback but not so common.

References i

- Batish, Rachel (2018). ***Voicebot and Chatbot design.*** Packt Publishing Birmingham.
- Cercas Curry, Amanda et Verena Rieser (juin 2018). **« #MeToo Alexa : How Conversational Systems Respond to Sexual Harassment ».** In : *Proceedings of the Second ACL Workshop on Ethics in Natural Language Processing*. Sous la dir. de Mark Alfano et al. New Orleans, Louisiana, USA : Association for Computational Linguistics, p. 7-14.
- — (sept. 2019). **« A Crowd-based Evaluation of Abuse Response Strategies in Conversational Agents ».** In : *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*. Sous la dir. de Satoshi Nakamura et al. Stockholm, Sweden : Association for Computational Linguistics, p. 361-366.

References ii

- Henderson, Peter et al. (2018). **« Ethical challenges in data-driven dialogue systems ».** In : *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, p. 123-129.
- Neff, Gina (2016). **« Talking to bots : Symbiotic agency and the case of Tay ».** In : *International journal of Communication*.
- Pearl, Cathy (2016). ***Designing voice user interfaces : Principles of conversational experiences.*** " O'Reilly Media, Inc."
- Wang, Xuewei et al. (juill. 2019). **« Persuasion for Good : Towards a Personalized Persuasive Dialogue System for Social Good ».** In : *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Sous la dir. d'Anna Korhonen, David Traum et Lluís Màrquez. Florence, Italy : Association for Computational Linguistics, p. 5635-5649.

References iii

 Worswick, Steve (juill. 2018). ***Ethics and Chatbots.*** Medium blog post.