

# End-to-End Task-Oriented Dialogue Systems

~ Nalin Kumar



Charles University  
Faculty of Mathematics and Physics  
Institute of Formal and Applied Linguistics



unless otherwise stated

# About Me

- Int. MSc. (Major: Mathematics | Minor: CS) — NISER (2016-2021)
  - Supervisor: Dr. Subhankar Mishra
  - Thesis: Neural Machine Translation for Low-Resource Languages
- MS in CS — Computational Linguistics — Charles University (Prague) (2021-24)
  - Supervisor: Mgr. Ondřej Dušek, Ph.D.
  - Thesis: Neural Models for Multilingual Natural Language Generation and Understanding
- Started PhD this Oct (2024-)
  - Supervisor: Mgr. Ondřej Dušek, Ph.D.
  - Thesis: Efficient Methods for Natural Language Generation Systems
- Contact me:
  - Email: [nkumar@ufal.mff.cuni.cz](mailto:nkumar@ufal.mff.cuni.cz)
  - LinkedIn: <https://www.linkedin.com/in/nalin-kumar-7b92aa174/>
  - Website(s): <https://ufal.mff.cuni.cz/nalin-kumar>, <https://knalin55.github.io/>

# Content

1. What are dialogue systems?
2. Basic Types
3. Modules — ASR, NLU, DM, **NLG**, TTS
4. End-to-end systems
5. Evaluation

LEEETs-Dial: Linguistic Entrainment in  
End-to-End Task-oriented Dialogue systems

## Introduction to Dialogue Systems

Slides taken from the course NPFL099 at UFAL,  
Charles University



[https://ufal.mff.cuni.cz/courses/npfl099/2023#lectures\\_](https://ufal.mff.cuni.cz/courses/npfl099/2023#lectures_)

## Paper Accepted at NAACL Findings 2024



<https://aclanthology.org/2024.findings-naacl.46/>

# What are Dialogue Systems

## Definition:

- A (spoken) dialogue system is a computer system designed to interact with users in (spoken) natural language

## Wide definition – covers lots of different cases

- “smart speakers” / phone OS assistants
- phone hotline systems (even tone-dial ones)
- in-car systems
- assistive technologies: therapy, elderly care, companions
- entertainment: video game NPCs, chatbots



# Example – Smart Speakers

- Google, Amazon, Apple & others, Mycroft: open-source
- Really good microphones
  - and not much else – they work online only
- Huge knowledge bases
  - Google: combined with web search
- Lots of domains programmed in, but all by hand
  - integration with a lot of services (calendar, music, shopping, weather, news...)
  - you can add your own (with limitations)
- Can keep some context
- Conversational capabilities limited



# Example – ChatGPT

- Huge neural network
  - ~10B parameters (details unknown)
- 1. Pretrained on web text
- 2. Trained on instructions + solutions
  - much more on point for the task
- 3. Trained on lots of human feedback
  - 👍👎 for different outputs
  - trained to be convincing
- Black box, hard to control
- Can answer anything, but no guarantee on accuracy
  - “Confidently” hallucinates



# Basic Types

## Task-oriented

- focused on completing a certain task/tasks
- booking restaurants/flights, finding bus schedules, smart home...

## Non-task-oriented

- chitchat – social conversation, entertainment
- getting to know the user, specific persona
- gaming the Turing test



# Communication Domains

- “domain” = conversation topic / area of interest
- traditional: single/**closed-domain**
  - one well-defined area, small set of specific tasks
  - e.g. banking system on a specific phone number
- **multi-domain**
  - basically joining several single-domain systems (Google/Alexa/Siri)
- **open-domain**
  - “responds to anything” – the goal, but now mostly chitchat-only





# Modes of Communication

- **text**
  - most basic/oldest
  - easiest to implement, most robust
  - not completely natural
- **voice**
  - more difficult, but can be more natural
  - emotions, tone, personality
  - easy to deploy over the phone
  - hands-free
- **multimodal**
  - voice/text + graphics
  - additional modalities: video – gestures, mimics; touch
  - most complex



# Modules

- traditional main DS pipeline:

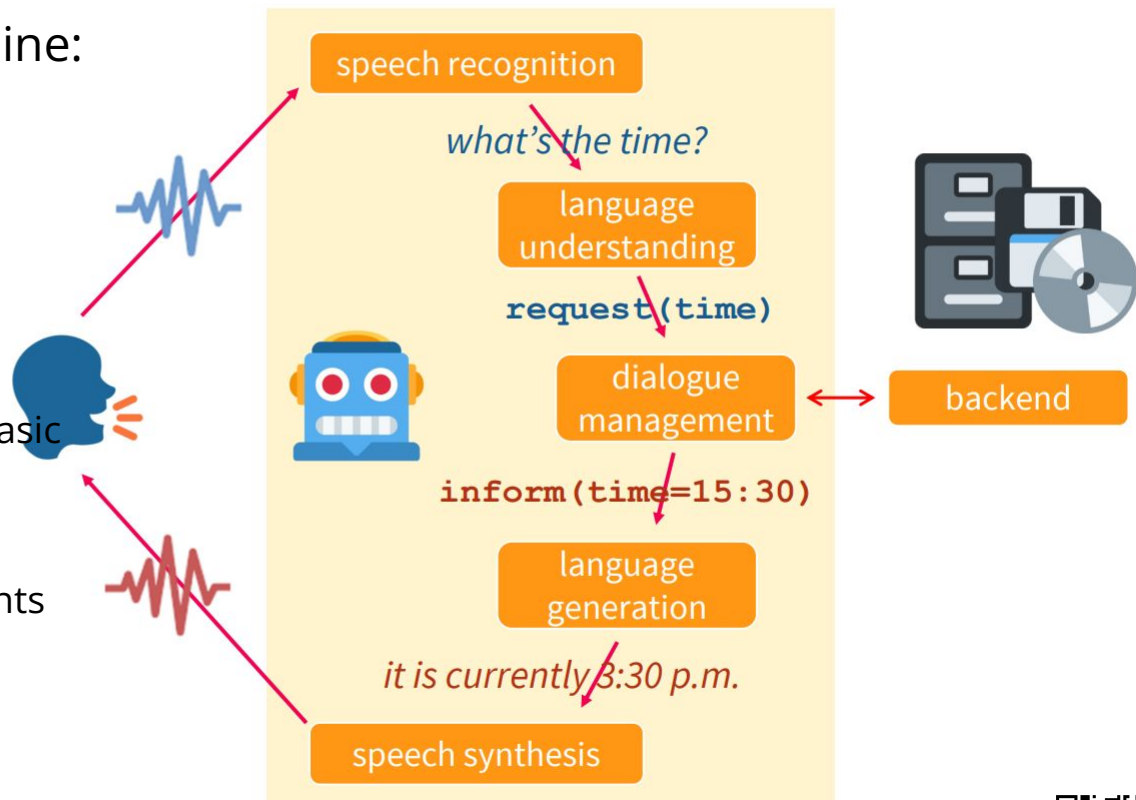
- voice → text
- text → meaning
- meaning → reaction
- reaction → text
- text → voice

- access to backend

- for anything better than basic chit-chat

- multimodal systems

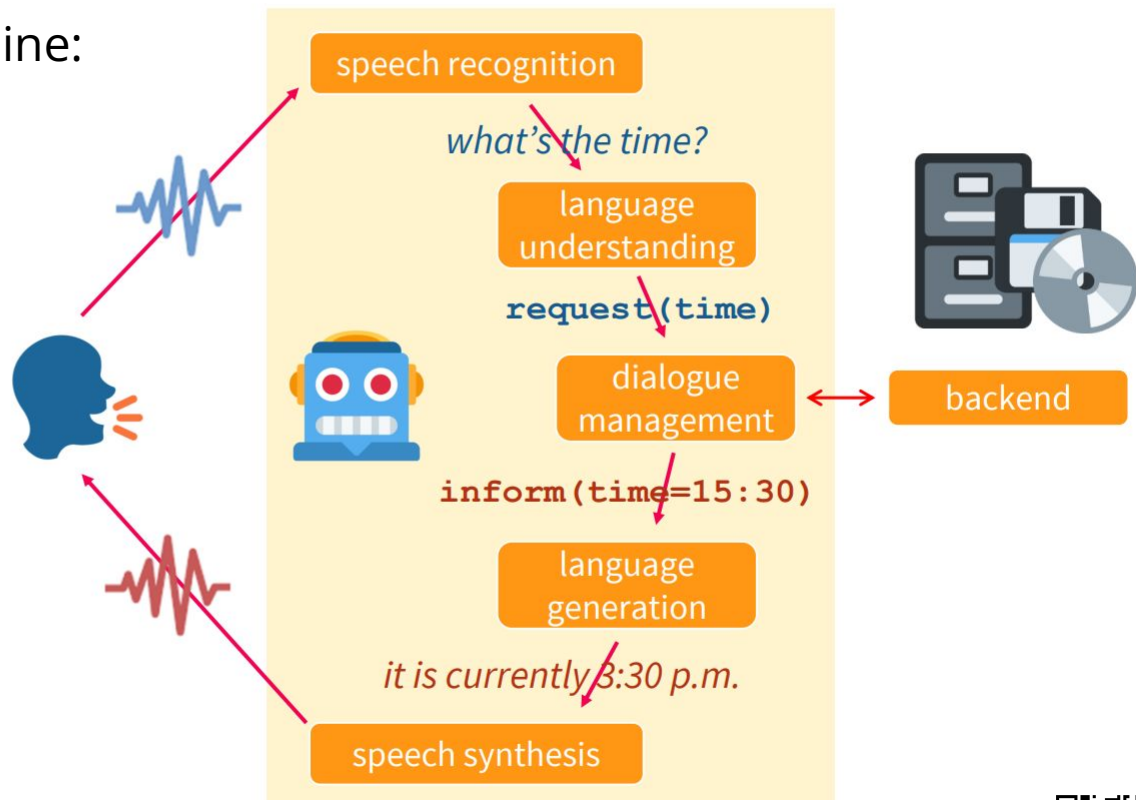
- need additional components



# Modules

- traditional main DS pipeline:

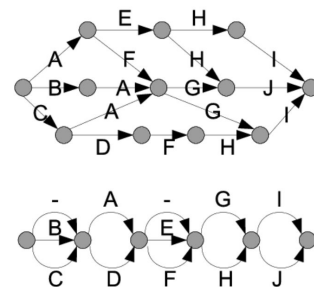
- **voice** → **text**
- text → meaning
- meaning → reaction
- reaction → text
- text → voice



# Automatic Speech Recognition (ASR)

- Converting speech signal (acoustic waves) into text
- Typically produces several possible hypotheses with confidence scores
  - n-best list
  - lattice
  - confusion network
- Very good in ideal conditions
- **Problems:**
  - noise, accents, longer distance, echo cancellation, channel (phone)...

0.8 I'm looking for a restaurant  
0.4 uhm looking for a restaurant  
0.2 looking for a rest tour rant



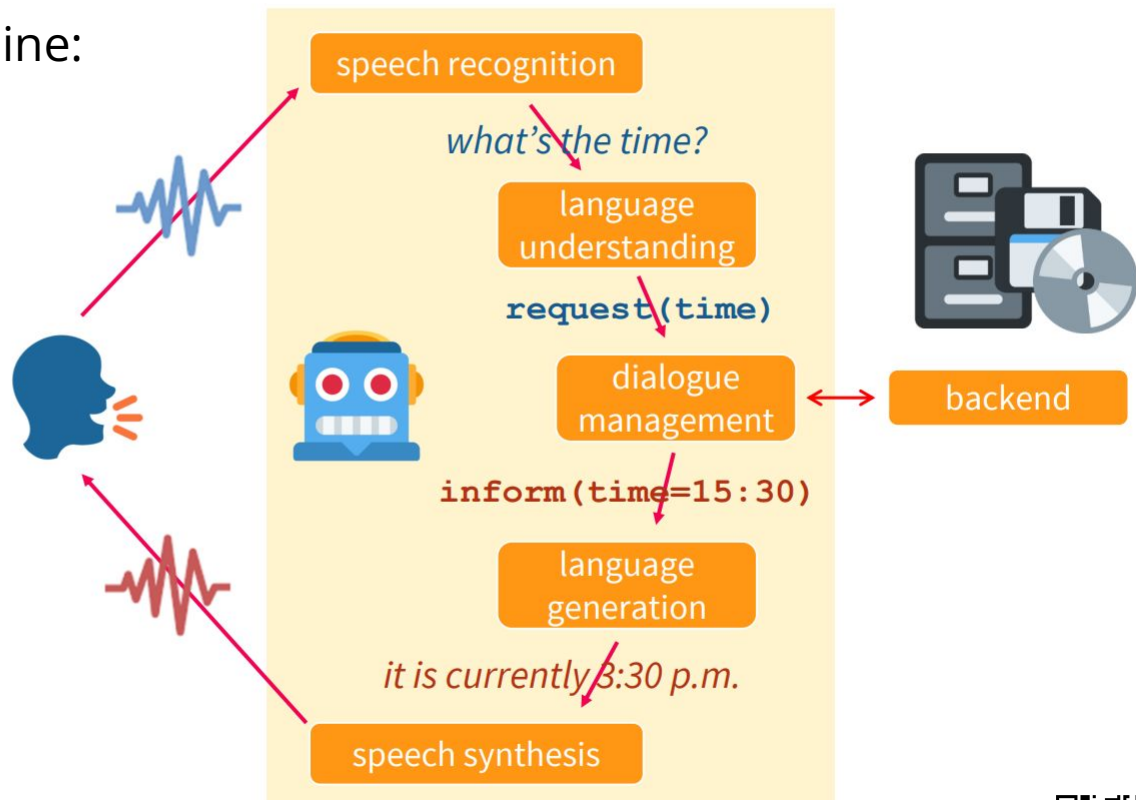
(Kazemian et al., ICMR 2008)  
<https://doi.org/10.1145/1460096.1460112>



# Modules

- traditional main DS pipeline:

- **voice** → **text**
- **text** → **meaning**
- meaning → reaction
- reaction → text
- text → voice



# Natural/Spoken Language understanding (NLU/SLU)

- Extracting the meaning from the (now textual) user utterance
- Converting into a structured semantic representation
  - **dialogue acts:**
    - act type/intent (inform, request, confirm)
    - slot/attribute (price, time...)
    - value (11:34, cheap, city center...)
    - typically intent detection + slot-value tagging
  - other, more complex – e.g. syntax trees
- Specific steps:
  - named entity resolution (NER)
    - identifying task-relevant names (London, Saturday)
  - coreference resolution
    - ("it" -> "the restaurant")

inform(food=Chinese, price=cheap)  
request(address)



# Natural/Spoken Language understanding (NLU/SLU)

- Problems:
  - recovering from bad ASR
  - ambiguities
  - variation

S: Leaving Baltimore. What is the arrival city?

U: fine Portland [ASR error]

S: Arriving in Portland. On what date?

U: No not Portland Frankfurt Germany

[On a Tuesday]

U: I'd like to book a flight from London to New York for **next Friday**

U: Chinese city center

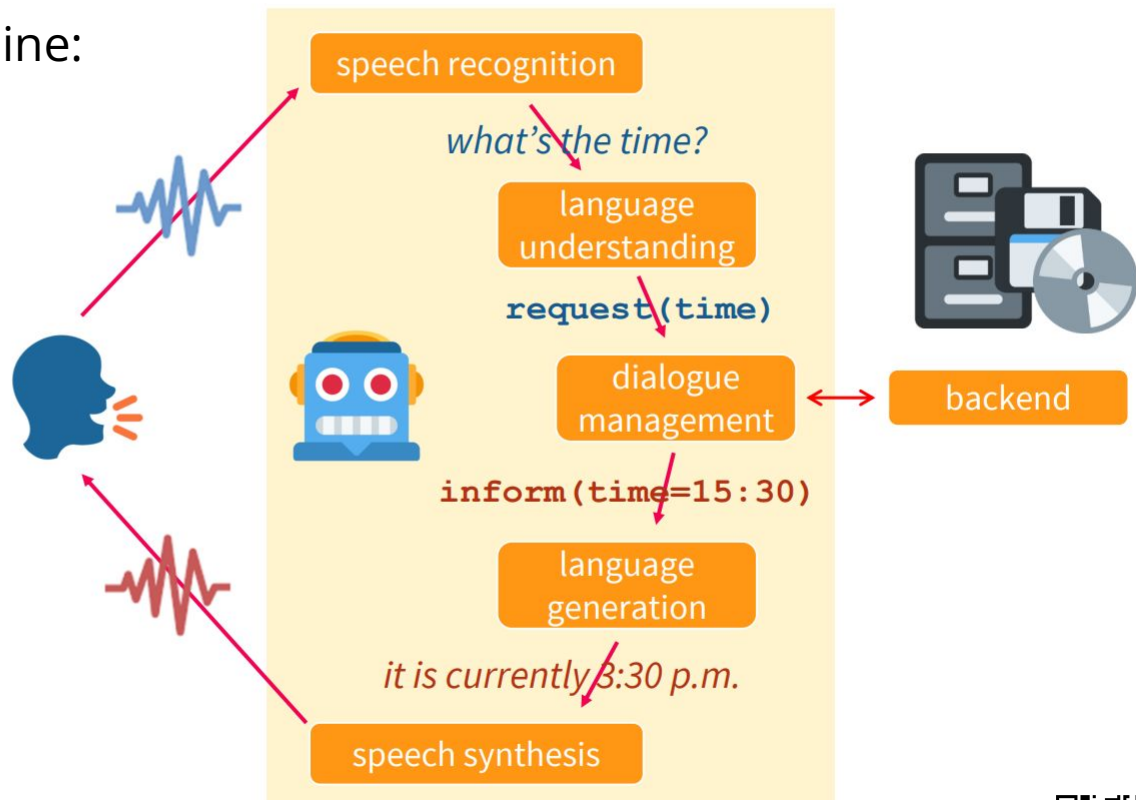
U: uhm I've been wondering if you could find me a restaurant that has Chinese food close to the city center please



# Modules

- traditional main DS pipeline:

- **voice** → **text**
- **text** → **meaning**
- **meaning** → **reaction**
- reaction → text
- text → voice





# Dialogue Manager (DM)

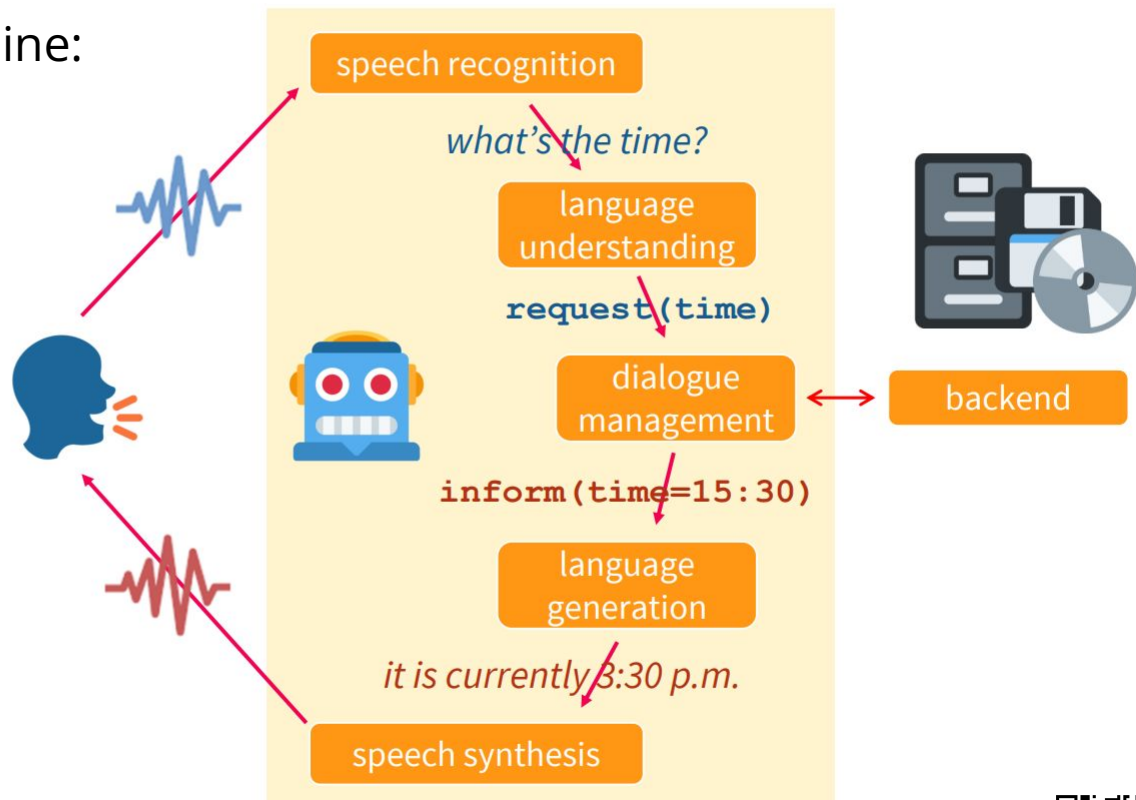
- Given NLU input & dialogue so far, responsible for deciding on next action
  - keeps track of what has been said in the dialogue
  - keeps track of user profile
  - interacts with backend (database, internet services)
- Dialogue so far = dialogue history, modelled by dialogue state
  - managed by dialogue state tracker
- System actions decided by dialogue policy



# Modules

- traditional main DS pipeline:

- **voice** → **text**
- **text** → **meaning**
- **meaning** → **reaction**
- **reaction** → **text**
- **text** → **voice**



# Natural Language Generation (NLG) / Response Generation

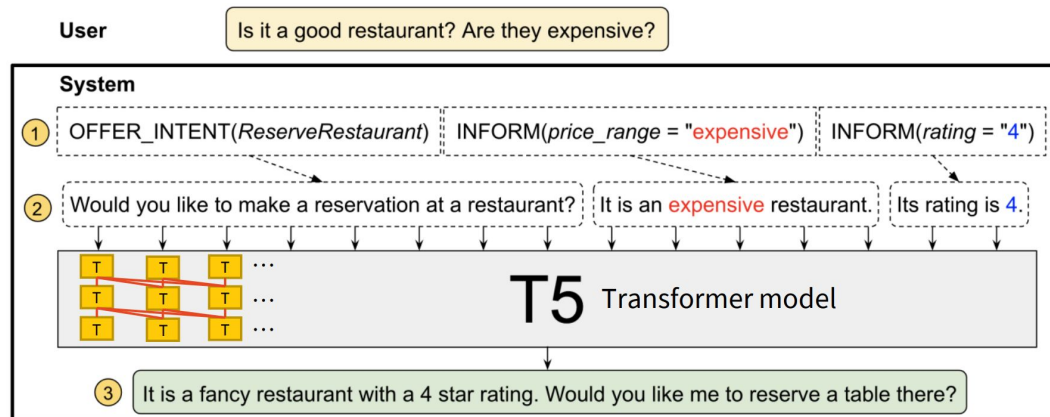
- Representing system dialogue act in natural language (text)
  - reverse NLU
- How to express things might depend on context
  - Goals: fluency, naturalness, avoid repetition (...)
- Traditional approach: templates
  - Fill in (=lexicalize) values into predefined templates (sentence skeletons)
  - Works well for limited domains

inform(name=Golden Dragon, food=Chinese, price=cheap)  
<name> is a <price>-ly priced restaurant serving <food> food  
Golden Dragon is a cheaply priced restaurant serving Chinese food.



# Natural Language Generation (NLG) / Response Generation

- Grammar-based approaches
  - grammar/semantic structures instead of templates
  - NLG realizes them (=converts to linear text) by applying syntactic transformation rules
- Statistical approaches
  - most prominent: neural networks (RNN/Transformer)
  - generating word-by-word
  - input: encoded semantics + previous words



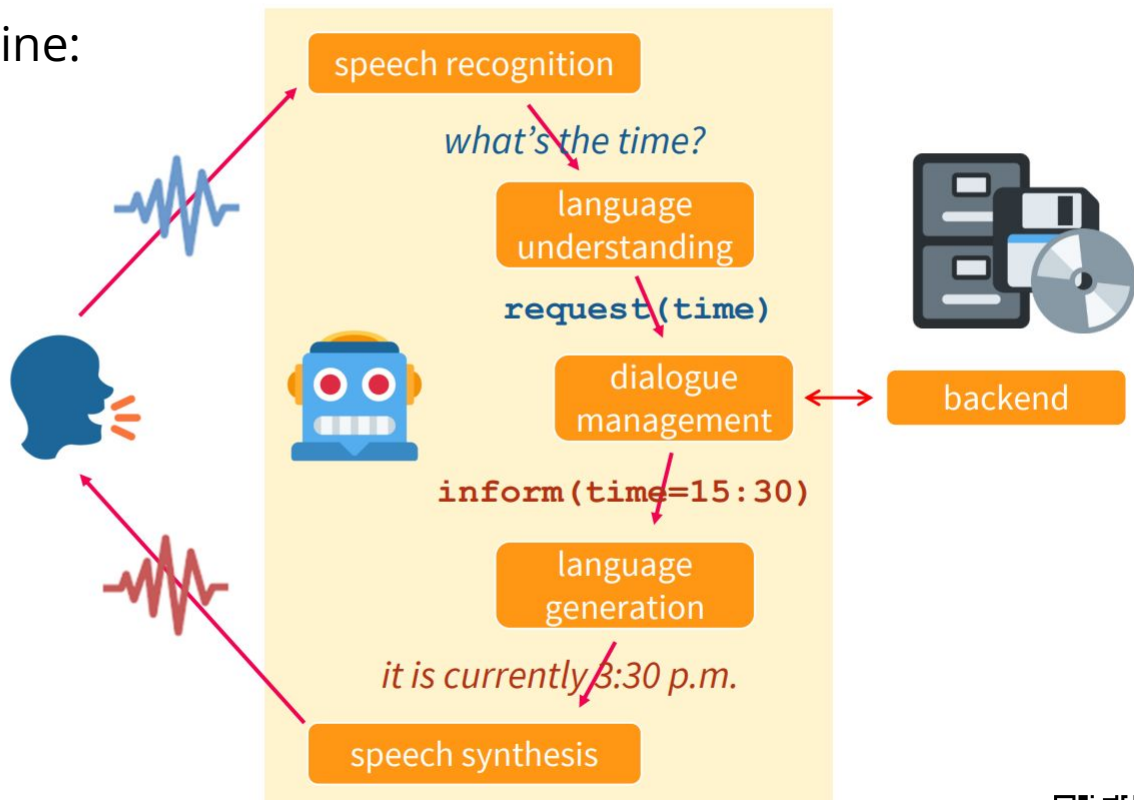
(Kale & Rastogi, 2020)  
<https://aclanthology.org/2020.emnlp-main.527/>



# Modules

- traditional main DS pipeline:

- **voice** → **text**
- **text** → **meaning**
- **meaning** → **reaction**
- **reaction** → **text**
- **text** → **voice**



# Text-to-speech (TTS) / Speech Synthesis

- Generate a speech signal corresponding to NLG output
  - text → sequence of phonemes
    - minimal distinguishing units of sound (e.g. [p], [t], [ŋ] “ng”, [ə] “eh/uh”, [i:] “ee”)
    - + pitch/intonation, speed, pauses, volume/accents •
- Standard pipeline:
  - text normalization
    - abbreviations
    - punctuation
    - numbers, dates, times
  - pronunciation analysis (grapheme → phoneme conversion)
  - intonation/stress generation
  - waveform synthesis

take bus number 3 at 5:04am  
take bus number three at five o four a m  
t eɪ k b ʌ s n ʌ m b ə θ r i: æ t f aɪ v ə u f o: r eɪ ə m



# Organizing the Components

- Basic pipeline:
  - $ASR \rightarrow NLU \rightarrow DM \rightarrow NLG \rightarrow TTS$
  - components oblivious of each other
- Joining the modules (experimental)
  - ASR + NLU
  - NLU + state tracking
  - NLU & DM & NLG



# End-to-End Systems

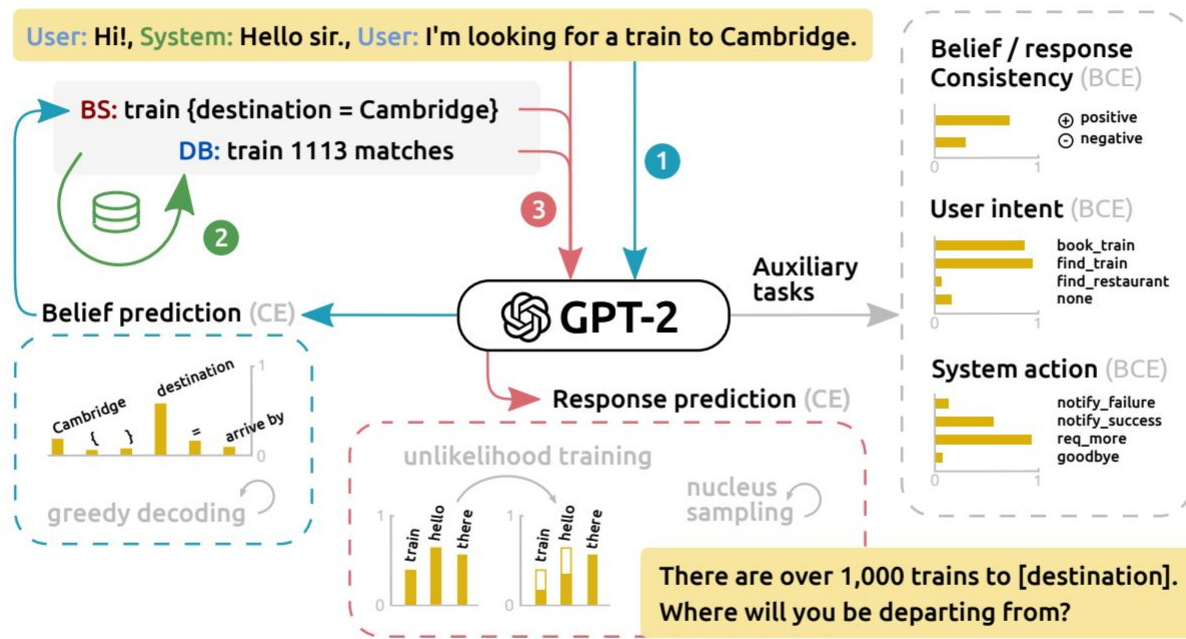
- typical for non-task-oriented
  - single network, trained e.g. on movie subtitles
- task oriented – experimental
- the whole system (NLU/DM/NLG) is a single neural network
  - joint training (“end-to-end”)
  - more elegant
  - potentially easily retrainable
- typically still needs annotation
  - same as individual modules
  - can be less predictable
- connecting the database is a problem





# AuGPT (Kulhánek et al., 2021)

- GPT2 based model
- two stages:
  - finetuned GPT-2 LM predicts a belief.
  - database results are obtained using the belief and everything is passed to the GPT-2 again to predict a final delexicalized response.
  - possible auxiliary tasks (belief consistency, intent classification, system action classification).



(Kulhánek et al., 2021)

<https://arxiv.org/pdf/2102.05126>



# Example

U:I want to find a moderately priced restaurant.

GBF:restaurant {price range = moderate}

D: restaurant 31 matches

GT:I have many options available for you! Is there a certain area or cuisine that interests you?

GTD:I have many options available for you! Is there a certain area or cuisine that interests you?

U:Yes I would like the restaurant to be located in the center of the attractions.

GBF:restaurant {area = centre, price range = moderate}

D: restaurant 21 matches

GT:There are 21 restaurants available in the centre of town. How about a specific type of cuisine?

GTD:There are 21 restaurants available in the [area] of town. How about a specific type of cuisine?

U:i need to know the food type and postcode and it should also have mutiple sports

GBF:restaurant {area = centre, price range = moderate}

D: restaurant 21 matches

GT:I am sorry I do not understand what you just said. Please repeat in a way that makes sense.

GTD:I am sorry I do not understand what you just said. Please repeat in a way that makes sense.

U:get me the food type and the post code

GBF:restaurant {area = centre, price range = moderate}

D: restaurant 21 matches

GT:My favorite it The Copper Kettle at 4 Kings Parade City Centre CB21SJ. It serves British food. Does that interest you?

GTD:My favorite it [name] at [address] [postcode]. It serves [food] food. Does that interest you?



# Evaluation

- Depends on dialogue system type / specific component
- Types:
  - **extrinsic** = how the system/component works in its intended purpose (ideal)
    - effect of the system on something outside itself, in the real world (i.e. user)
  - **intrinsic** = checks properties of systems/components in isolation,
    - self-contained
  - **subjective** = asking users' opinions,
    - e.g. questionnaires (~manual/human)
    - should be more people, so overall not so subjective
    - objective = measuring properties directly from data (~automatic)
    - might or might not correlate with users' perception
- Evaluation discussed here is mostly quantitative
  - i.e. measuring & processing numeric values
  - (qualitative ~ e.g. in-depth interviews, more used in social science)



# Subjective Evaluation: Getting Subjects

- Can't do without people
  - simulated user = another (simple) dialogue system
    - can help & give guidance sometimes, but it's not the real thing – more for intrinsic
- In-house = ask people to come to your lab (or access your website)
  - students, friends/colleagues, hired people
  - expensive, time-consuming
  - doesn't scale (difficult to get subjects)
- Crowdsourcing = hire people over the web
  - much cheaper, faster, scales (unless you want e.g. Czech)
  - not real users – mainly want to get their reward
- Real users = deploy your system and wait
  - best, but needs time & advertising & motivation
  - you can't ask too many questions
  - Ethics and privacy implications



# Subjective Evaluation — Some examples

- **Success rate** (task-oriented):
  - Did you get all the information you wanted?
  - typically different from objective measures!
- **Future use:**
  - Would you use the system again?
- **Likeability/engagement:**
  - Did you enjoy the conversation?
- **ASR/NLU:**
  - Do you think the system understood you well?
- **NLG:**
  - Were the system replies fluent/well-phrased?
- **TTS:**
  - Was the system's speech natural?



# Intrinsic Objective Evaluation: NLU

- Slot Precision & Recall & F-measure (F1)

true: inform(name=Golden Dragon,  
food=Chinese)  
NLU: inform(name=Golden Dragon,  
food=Indian, price=high)

precision

$$P = \frac{\text{\#correct slots}}{\text{\#detected slots}}$$

how much of the identified stuff  
is identified correctly

recall

$$R = \frac{\text{\#correct slots}}{\text{\#true slots}}$$

how much of the true stuff  
is identified at all

F-measure

$$F = \frac{2PR}{P + R}$$

harmonic mean – you want both  $P$  and  $R$   
to be high (if one of them is low, the mean is low)



# Extrinsic / Intrinsic Objective Evaluation: Dialogue Manager

- Objective measures (task success rate, duration) can be measured with a user simulator
  - works on dialogue act level
  - responds to system actions
- Simulator implementation
  - handcrafted (rules + a bit of randomness)
  - n-gram models over DA/dialogue turns + sampling from distribution
  - agenda-based (goal: constraints, agenda: stack of pending DAs)
  - reinforcement learning policy
- Problems:
  - cost: the simulator is basically another dialogue system
  - might not be fair (depending on the simulation accuracy)
    - typically your system would work better with a simulator than with humans



# Extrinsic / Intrinsic Objective Evaluation: NLG

- No single correct answer here
  - many ways to say the same thing
- **Word-overlap** with reference text(s): **BLEU** score

range [0,1]  
(percentage)

$BLEU = BP \cdot \exp \left( \sum_{n=1}^4 \frac{1}{4} \log(p_n) \right)$

geometric mean

**brevity penalty** (1 if output longer than reference,  
goes to 0 if too short)

**n-gram precision:**  
 $p_n = \frac{\sum_u \# \text{ matching n-grams in } u}{\sum_u \# \text{ n-grams in } u}$

- **n-gram** = span of adjacent n tokens
  - 1-gram (one word) = unigram,
  - 2-gram (2 words) = bigram,
  - 3-gram = trigram
- Not reliable — still used though
- Alternatives: diversity, slot-error





# Extrinsic Objective Evaluation

Metrics:

- **Task success** (task-oriented):
  - did the user get what they wanted?
  - testers with agenda → check if they found what they were supposed to
  - [warning] sometimes people go off script
  - basic check: did we provide any information at all? (any bus/restaurant)
- **Duration: number of turns**
  - task oriented: fewer is better,
  - non-task-oriented: more is better
- **Other (not so standard):**
  - % returning users
  - % turns with null semantics (task-oriented)
  - % swearing / thanking



# Summary

1. **What are dialogue systems?**
  2. Basic Types
  3. Modules — ASR, NLU, DM, **NLG**, TTS
  4. End-to-end systems
  5. Evaluation
- system designed to interact with users



[https://www.shutterstock.com/shutterstock/photos/2029791242/display\\_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg](https://www.shutterstock.com/shutterstock/photos/2029791242/display_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg)



# Summary

1. What are dialogue systems?
  2. **Basic Types**
  3. Modules — ASR, NLU, DM, **NLG**, TTS
  4. End-to-end systems
  5. Evaluation
- Task-oriented — booking systems
  - Non task-oriented — chitchat



[https://www.shutterstock.com/shutterstock/photos/2029791242/display\\_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg](https://www.shutterstock.com/shutterstock/photos/2029791242/display_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg)



# Summary

1. What are dialogue systems?
2. Basic Types
3. **Modules — ASR, NLU, DM, NLG, TTS**
4. End-to-end systems
5. Evaluation

- ASR — voice → text
- NLU — text → meaning
- DM — meaning → reaction
- NLG — reaction → text
- TTS — text → voice



[https://www.shutterstock.com/shutterstock/photos/2029791242/display\\_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg](https://www.shutterstock.com/shutterstock/photos/2029791242/display_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg)



# Summary

1. What are dialogue systems?
2. Basic Types
3. Modules — ASR, NLU, DM, NLG, TTS
4. **End-to-end systems**
5. Evaluation

- Typical for non task oriented
- Not so trivial of task-oriented
  - AuGPT
    - Given dialogue history, predicts belief state
    - Given dialogue history + belief state, predicts response



[https://www.shutterstock.com/shutterstock/photos/2029791242/display\\_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg](https://www.shutterstock.com/shutterstock/photos/2029791242/display_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg)



# Summary

1. What are dialogue systems?
2. Basic Types
3. Modules — ASR, NLU, DM, NLG, TTS
4. End-to-end systems
5. **Evaluation**

- extrinsic
  - how good the system works in its intended purpose (whole)
- intrinsic
  - how good the submodules work in isolation
- subjective
  - taking users' opinions



[https://www.shutterstock.com/shutterstock/photos/2029791242/display\\_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg](https://www.shutterstock.com/shutterstock/photos/2029791242/display_1500/stock-vector-illustration-of-a-man-taking-a-deep-breath-2029791242.jpg)



# LEEETs-Dial: Linguistic Entrainment in End-to-End Task-oriented Dialogue systems

Nalin Kumar, Ondřej Dušek

Access full paper here



<https://aclanthology.org/2024.findings-naacl.46/>



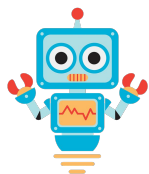
Charles University  
Faculty of Mathematics and Physics  
Institute of Formal and Applied Linguistics



unless otherwise stated

# Linguistic Entrainment

I would like a taxi from  
Saint John's college to  
Pizza Hut Fen Ditton.



What time do you want to  
leave?



I would like a taxi from  
Saint John's college to  
Pizza Hut Fen Ditton.



What time would you like to  
leave from Saint John's  
college?





# Linguistic Entrainment

- Conversational partners adapt to each other
- Entrainment = alignment / adaptation at various levels:
  - Phonetic patterns
  - Syntactic constructions
  - Lexical terms
- In context of dialogue system (DS), entrainment improves user experience
- Existing works in task-oriented DS
  - Task success rate
  - Low diversity outputs (content planning ↑, surface realization ↓)
  - Lack of dedicated support for user-system dialogue alignment



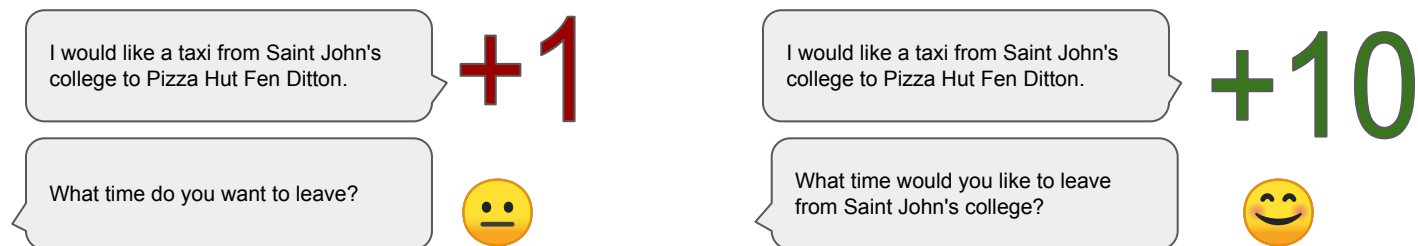
# Objective

- Entrainment for end-to-end task-oriented DS
- Primary focus on lexical entrainment
- Three approaches:
  - **Instance weighting** — Data-centric — promote “nice” data instances
  - **User Likelihood Loss** — Additional loss — maximize probability of user tokens
  - **Keyword-based conditioning** — Additional hints concatenated to input — assisting model with “important” user tokens



# Instance Weighting (IW)

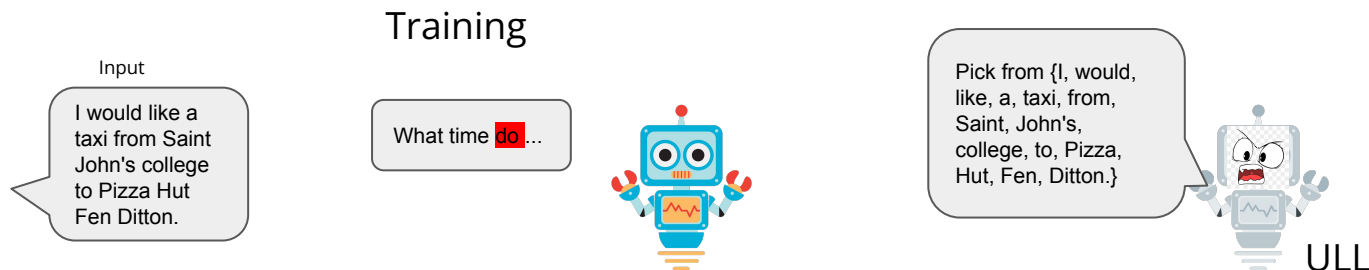
## Training



- Data-centric approach: promote “nice” data instances
  - “nice” = better 1-gram precision(user, system)
- Better 1-gram precision  $\rightarrow$  More system-user overlap  $\rightarrow$  Higher weight
- Two weight functions
  - Discrete ( $IW_1$ ): Less overlap ( $p < 0.25$ ) = 1 | More overlap ( $p \geq 0.25$ ) = 10
  - Continuous ( $IW_2$ ): Sigmoid-like function, with values ranging from 1-10



# User Likelihood Loss (ULL)

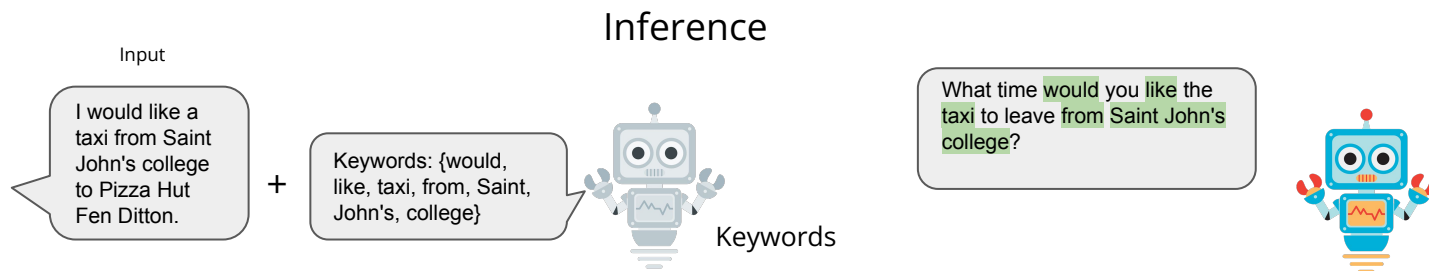


- Recall: need more overlap between user & system
- Additional loss: maximize probability of user tokens
- Minimize  $L_t(p(.|x_{<t}), U) = -\alpha \cdot \log \left( \sum_{u \in U} p(u|x_t) \right)$

U = set of user tokens



# Conditioning Generation on Lexical Keywords (LK)



- Keyword-based conditioning: Assisting model with “important” user tokens
  - “<|Keyword|> KEYWORDS\_LIST” added to model input
- Training:
  - Keyword = overlapping words in user-system
- Inference:
  - Keyword = select user tokens using self-attention scores
- Blending parameter  $\sigma$ 
  - Expose model training to inference keywords with probability  $\sigma$
  - $\sigma = 0, 0.5, 0.05$



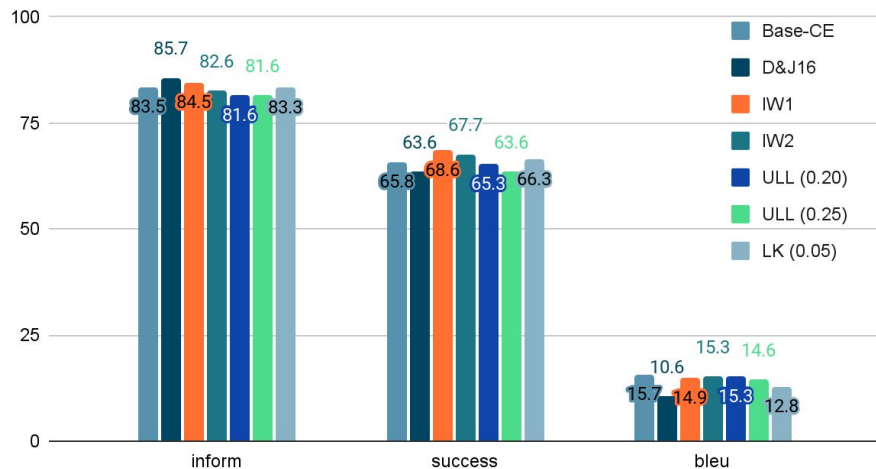
# Data & Training Setup

- MultiWOZ 2.1 dataset
  - 10k task-oriented human-human written dialogues spanning over 7 domains.
- Base model: AuGPT (Kulhánek et al., 2021)
  - end-to-end GPT-2 based model
- In addition to cross-entropy loss (CE), we also experiment with Unlikelihood Loss (UnL)
- Baselines
  - Base-Ce: Vanilla Base model without unlikelihood loss
  - D&J16: Base-Ce + reranking (function based on precision scores) while decoding
- Evaluation Metrics:
  - Standard MultiWOZ metrics for response generation and state tracking
  - 1-gram precision and recall ( $\text{lex-p}_1$ ,  $\text{lex-r}_1$ ) for lexical entrainment
  - 2,3-gram precisions on POS tags ( $\text{syn-p}_2$ ,  $\text{syn-p}_3$ ) for syntactic entrainment

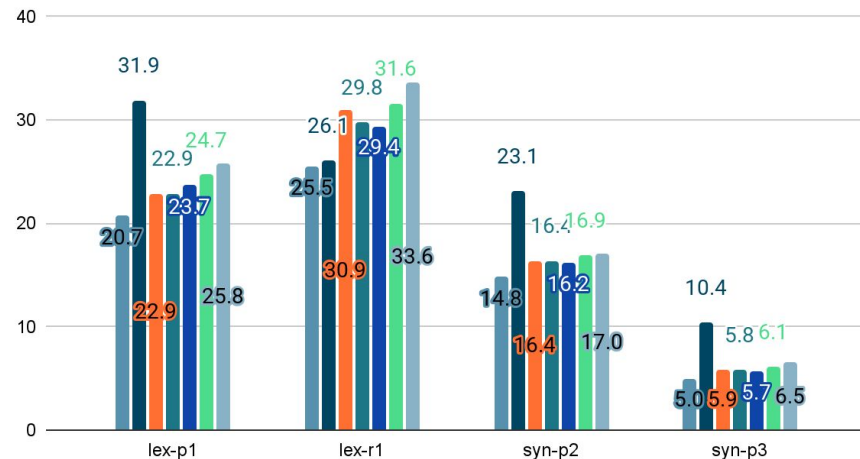


# Automatic Evaluation Metrics

MultiWOZ metrics



Linguistic Entrainment metrics

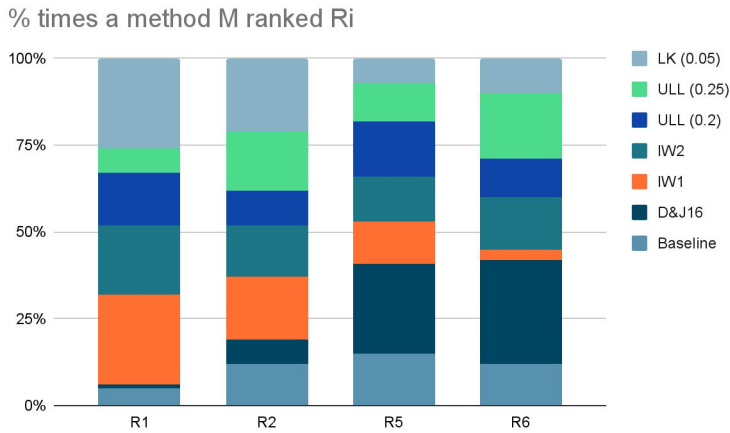


- MultiWOZ metrics: similar to baseline (slight drop for ULL)
- Entrainment scores: LK — best recall, D&J16 — best precision



# Manual Evaluation

Model	Mean Rank (↓)
Baseline	4.18
D&J16	5.35
IW <sub>1</sub>	<b>3.16</b>
IW <sub>2</sub>	3.77
ULL (0.20)	4.17
ULL (0.25)	4.33
LK	3.25



- Relative ranking - based on naturalness
- Mean Rank
  - IW<sub>1</sub> — natural-looking outputs, LK — better entrained outputs
  - D&J16 — shorter, less polite, less interactive outputs
- In general, slightly inconsistent outputs from alignment-based models
  - Some outputs excellent; some have hallucinations and repetitions.





# Conclusion

- Entrainment – outputs look more natural – better user experience
- Three approaches
  - Instance weighting: better task success rate + fluent
  - User likelihood loss: fragile, hallucinates with higher  $\alpha$  values
  - Keyword-based conditioning: fluent + better entrainment
- Future directions
  - Syntactic entrainment
  - Retrieval-augmented generation based approaches



# LEETs-Dial: Linguistic Entrainment in End-to-End Task-oriented Dialogue systems



`nkumar@ufal.mff.cuni.cz`



`odusek@ufal.mff.cuni.cz`

Follow our work at <https://github.com/knalin55/LEETs-Dial>



## Contact me:

- Email: nkumar@ufal.mff.cuni.cz
- LinkedIn: <https://www.linkedin.com/in/nalin-kumar-7b92aa174/>
- Website(s): <https://ufal.mff.cuni.cz/nalin-kumar>, <https://knalin55.github.io/>

## Questions?