The patience of robots

Linguistic complexity and cognitive workload: measurement and management, day 5, ESSLLI 2014

Asad Sayeed

Uni-Saarland

The answer is still 42.

Marvin: "And then of course I've got this terrible pain in all the diodes down my left side."

Arthur: "Is that so?"

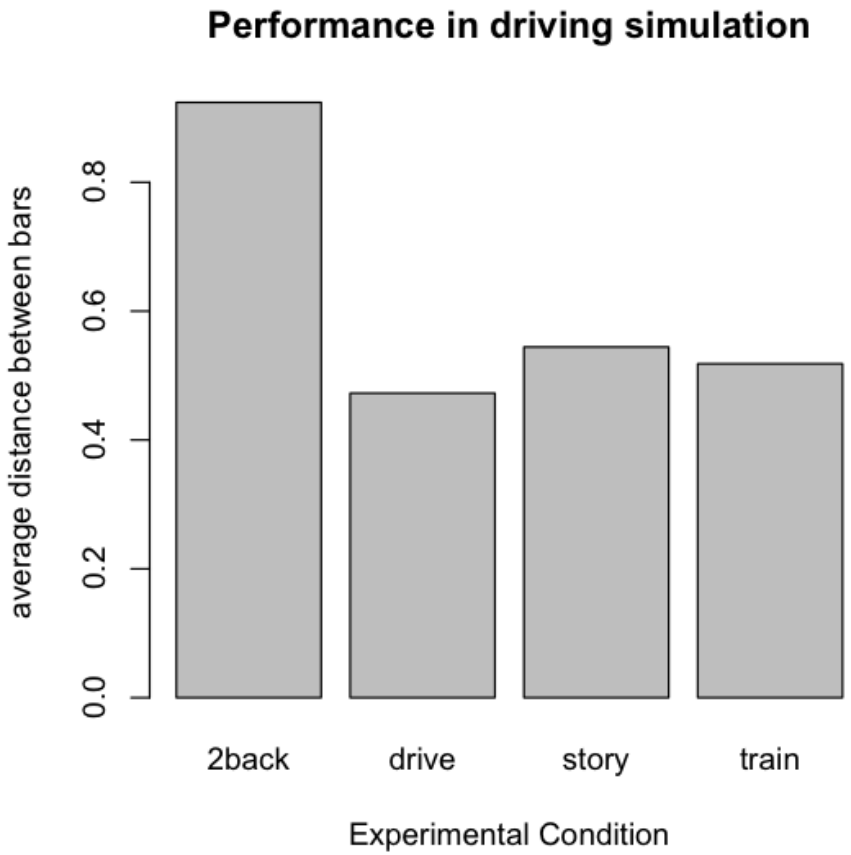
Marvin: "Oh yes. I mean I've asked for them to be replaced, but no one ever listens."

Arthur: "I can imagine"

— *The hitchhiker's guide to the galaxy*, Douglas Adams

Before we begin, RESULTS!

It turned out three was enough.



- Three participants.
- Significant differences: driving vs. driving+story, driving+story vs. driving+n-back.

At the very least, workload effects are pretty strong!

On to the main event!

How did we get here, again? Time for a recap.

This is what we've done so far.

- We discussed the overall social/user interface issue with dual-task activities.
  - People-people interactions show that language matters.
  - We focused on driving.
- We talked about linguistic complexity.
  - How do we quantify the "work done" in language use?
  - Different ways of going about it, but most successful recent effort is information-theoretic.

This is what we've done so far.

- Then we talked about workload measurement in terms of psychological experimentation.
  - A big variety of "extrinsic" vs. "intrinsic" measures.
  - Extrinsic measures defined in terms of task-performance (e.g. driving deviation, steering wheel reversal).
  - Intrinsic measures tend to be more physiological (e.g. pupil diameter, skin conductance).
- Yesterday, we discussed individual differences and adaptation to the individual user.

But that leaves us with a question.

Exactly HOW would we adapt dialogue systems to the user/task combo?

This is where we get kind of "aspirational".

What are the elements we have to combine?

The dialogue system needs to take into account:

- User characteristics (cognitive capacity, interests).
- "Regulatory"/safety priorities.
- Primary task performance (e.g. flight booking).
- "Secondary" task performance (see safety).
- Exterior environment.
- Linguistic complexity.

And they're interdependent.

That's a tall order!

But let's break it down into research questions.

There are two types:

- What is the relationship between observed workload interactions and the architecture of the mind?
- What does this imply about the engineering of dialogue systems?

But let's break it down into research questions.

There are two types:

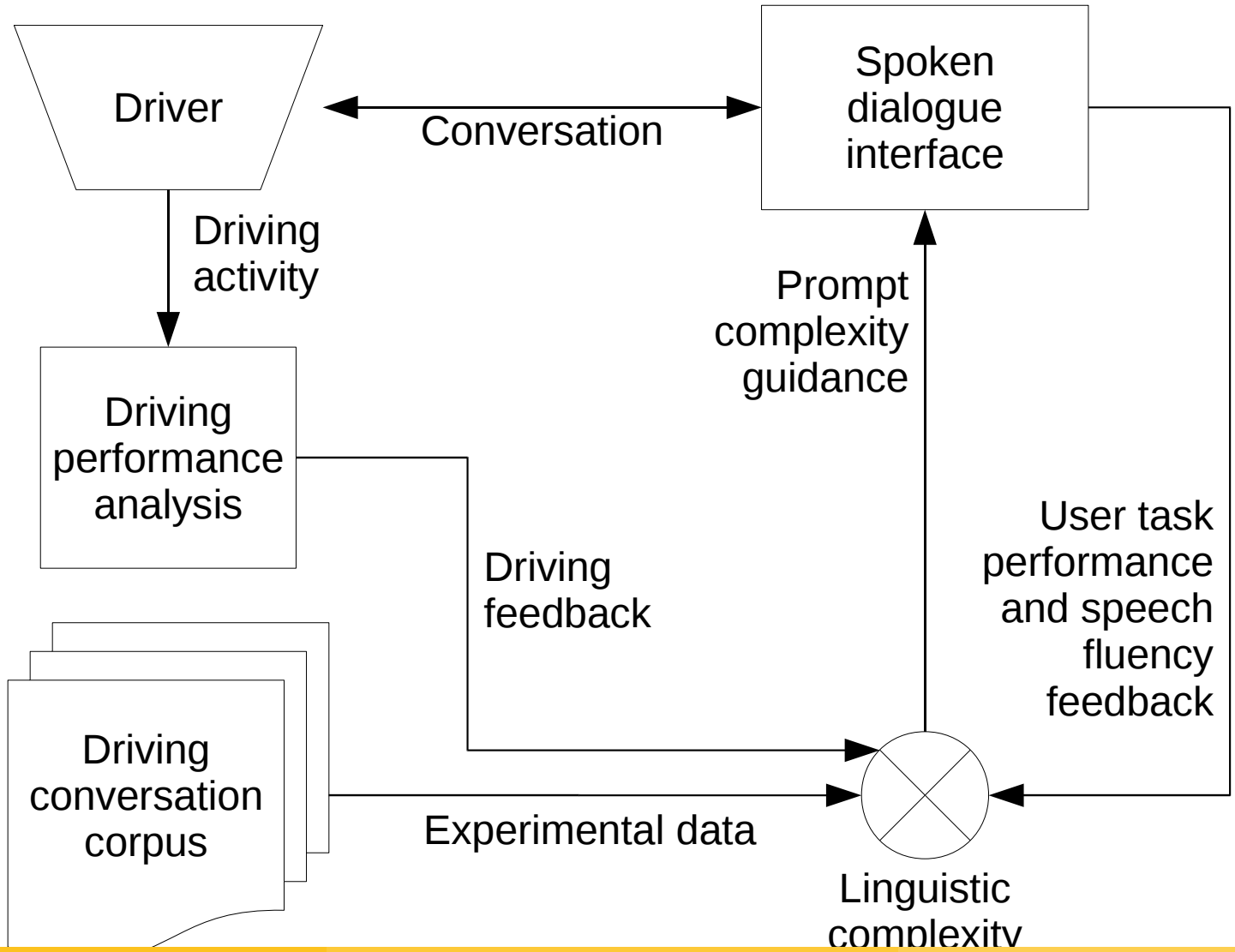
- What is the relationship between observed workload interactions and the architecture of the mind?
- What does this imply about the engineering of dialogue systems?

We've explored some preliminary answers to the former question, so today we'll focus on the latter.

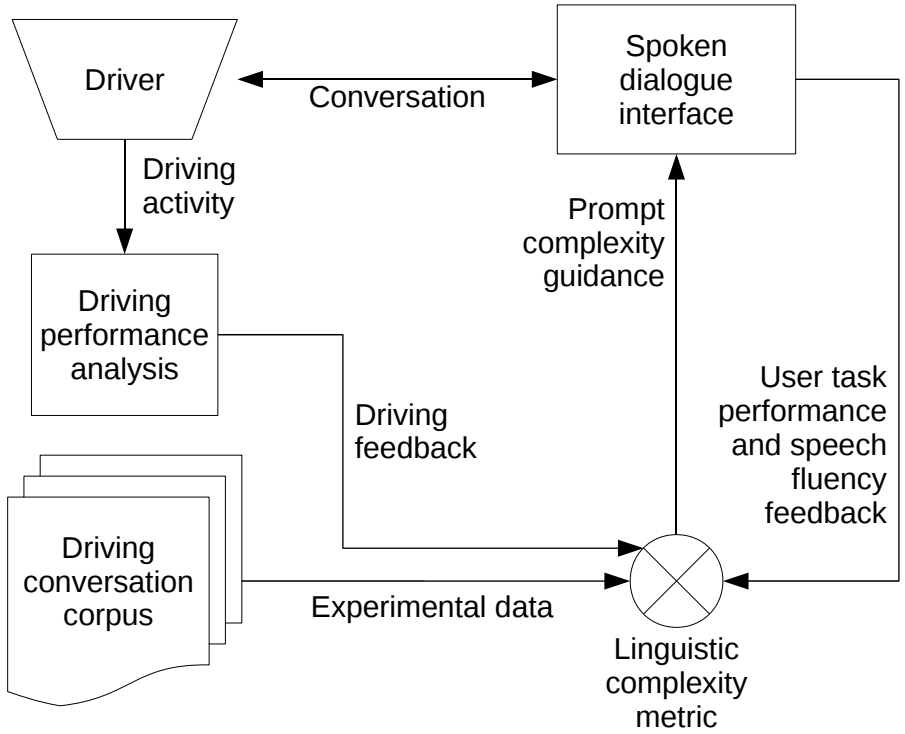
So, driving.

We've talked about driving, language, and driving and language. What would a workload-adaptive system actually look like?

Maybe something like this (Demberg and Sayeed, 2011).



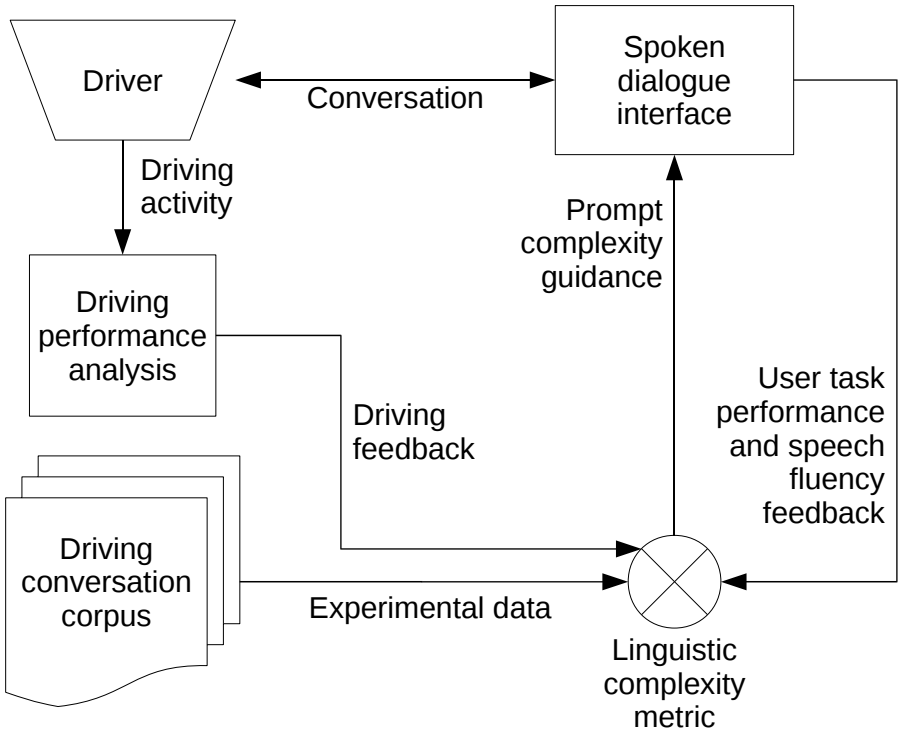
Let's pay attention to some key components.



The linguistic complexity metric.

- Let's say **for now** that surprisal is a good candidate.

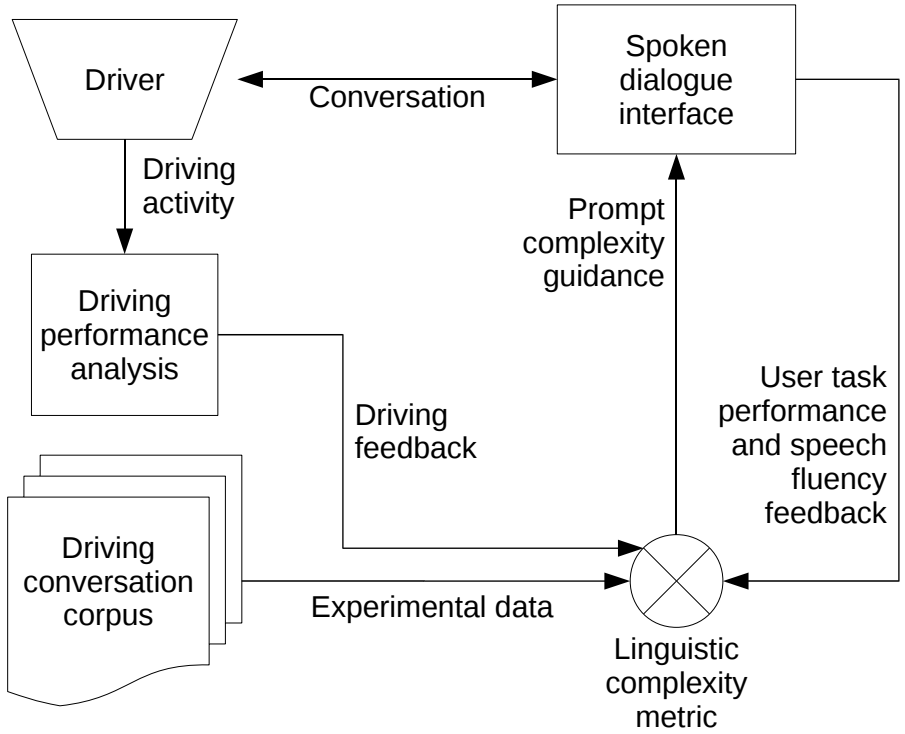
Let's pay attention to some key components.



Driving performance feedback.

- There are some "objective" measures we can throw in here – wheel reversal, distance from cars ahead, etc.

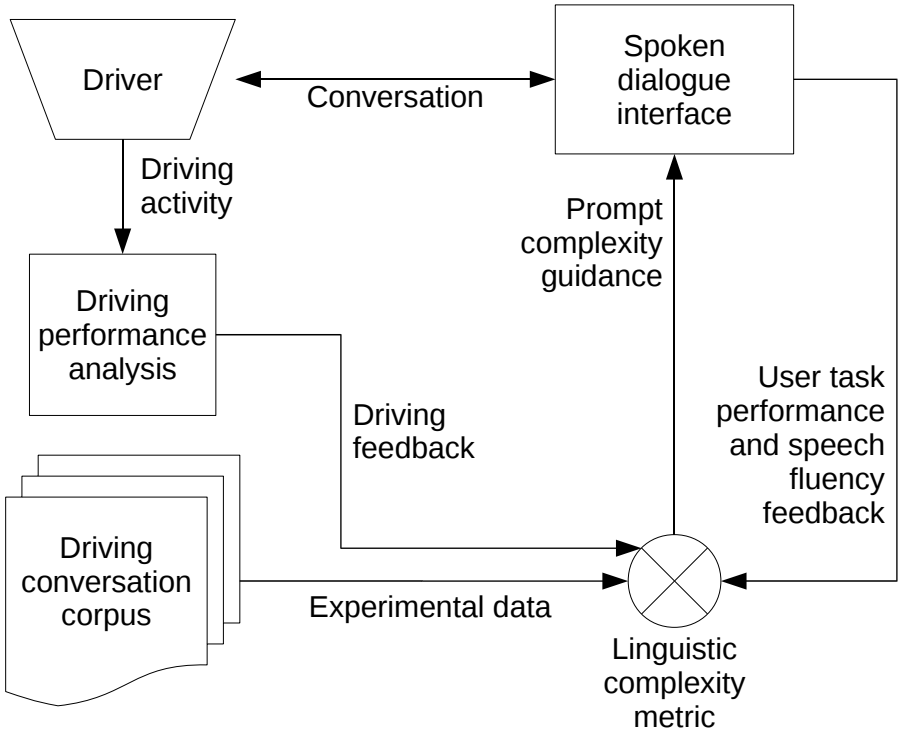
Let's pay attention to some key components.



User task performance.

- Did you manage to book your flight? Heh.

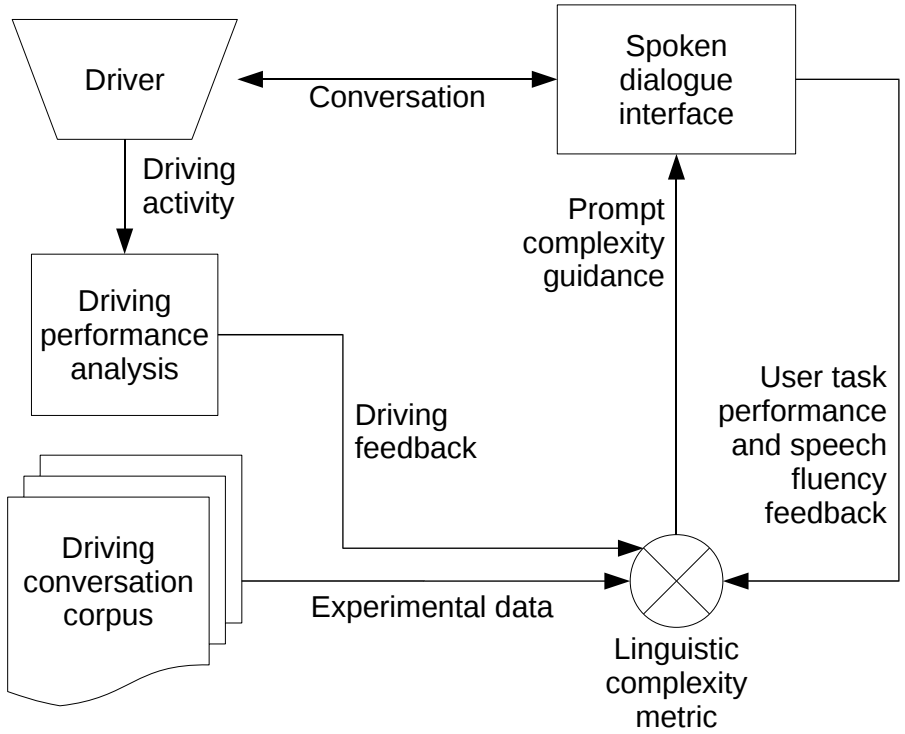
Let's pay attention to some key components.



User task performance.

- Did you manage to book your flight? Heh.

Let's pay attention to some key components.



Prompt complexity guidance.

- Here we have a problem: **what exactly are we trying to guide?**

This is the bit that gets aspirational.

Dialogue systems, in pieces.

The different portions of a dialogue system:

- **ASR** Automatic speech recognition: handles input acoustics, phonetics, phonology, etc.
- **NLU** Natural language understanding: handles the syntax, semantics, pragmatics of the spoken input.
- **CS** Content selection: determines the semantic/pragmatic content of the response to the user request.

Dialogue systems, in pieces.

The different portions of a dialogue system:

- NLG: Natural language generation: converts the content representation into the textual part of human language output (i.e. handles syntax/lexicon).
- TTS: Text-to-speech: handles the output phonology, phonetics, acoustics.

And you can think of them as a pipeline mediated by an overall Dialogue Management (DM) system.

Are they all equally relevant to workload management?

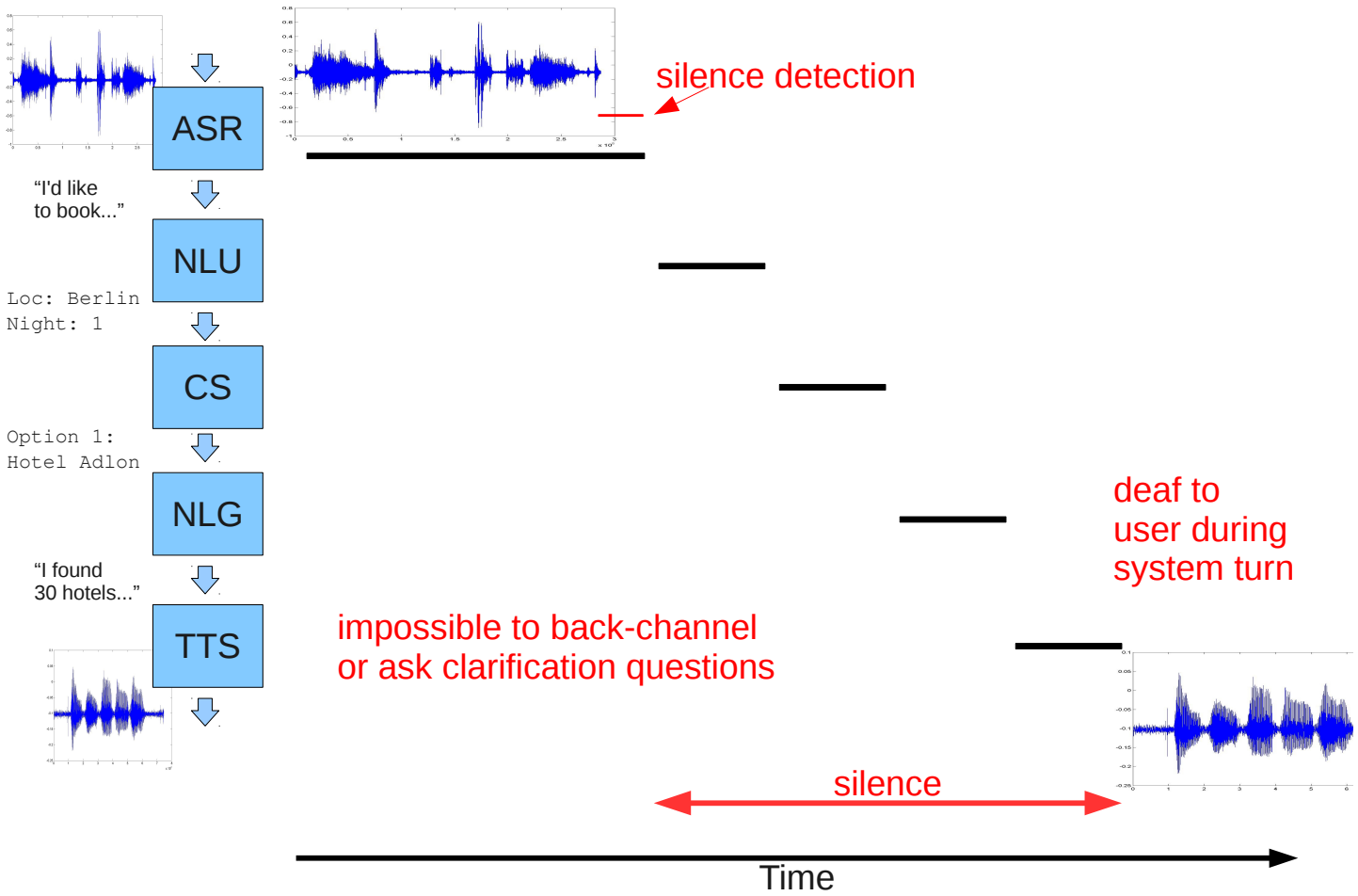
Probably not.

- Properly incremental NLU is necessary for rapid reaction to user requests.
- But most important for controlling workload:
  - CS — needs to select and schedule content relative to users overall use context.
  - NLG — needs to deal with the "classical" problems of linguistic complexity, esp. syntactic.

ASR and TTS are by NO means irrelevant here, however.

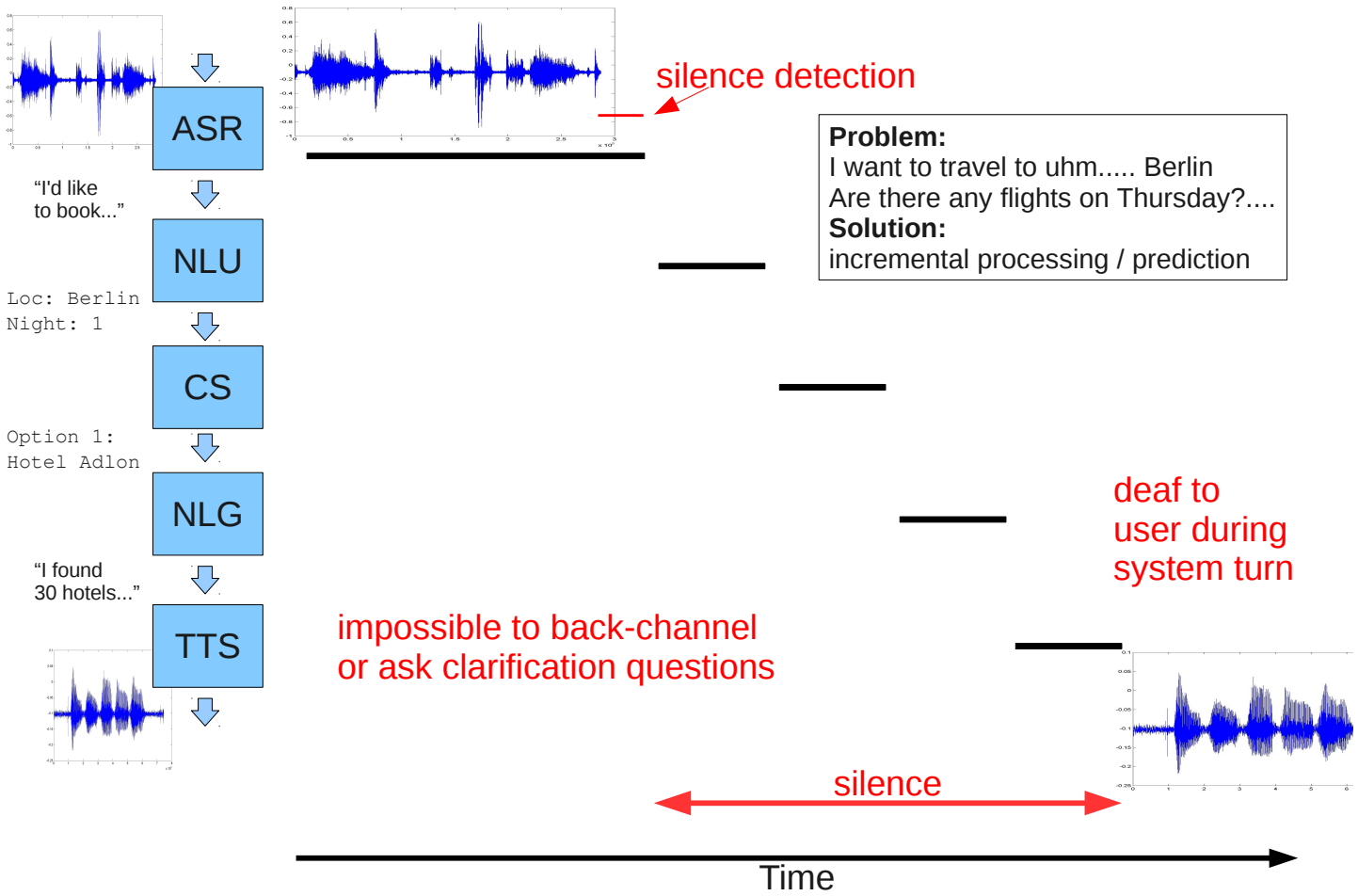
So, NLU and incrementality.

Processing in a traditional Spoken Dialogue System



So, NLU and incrementality.

Processing in a traditional Spoken Dialogue System



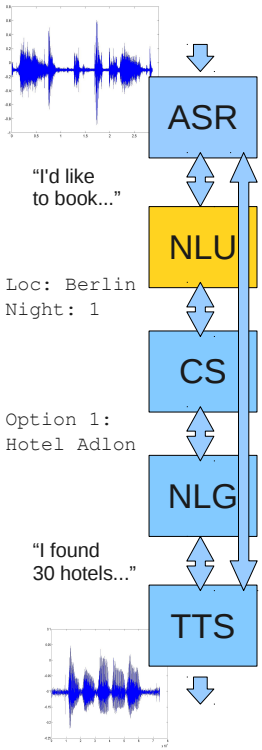
Human-human communication doesn't work that way.

We don't wait until the end of the sentence to start processing!

- Humans don't have long pauses to respond.
  - But even today, automatically consulting a remote system (say, TripAdvisor) is not instantaneous in practice!
- Humans interrupt all the time.
- Humans provide active feedback.

Remember, human cooperation $\Rightarrow$ lower workload.

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (traditional)

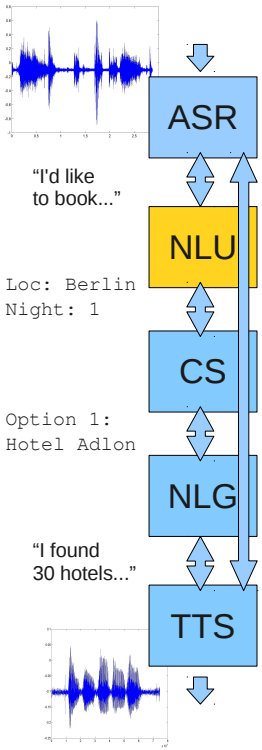
I'd like to book a double room for 2 adults in Berlin for one night.

→

Frame (traditional)

city:	Berlin
arrival:	
departure:	
duration:	1
# of people:	2
room type:	double

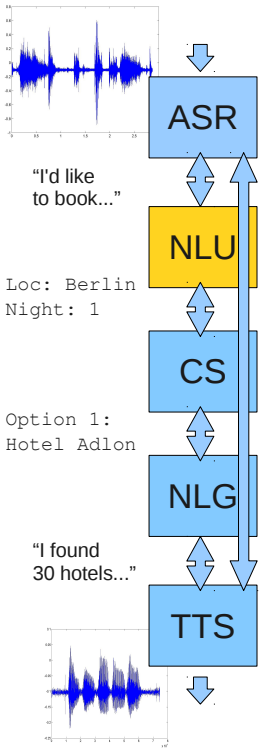
So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)	→	Frame (incremental)
I'd like		

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)

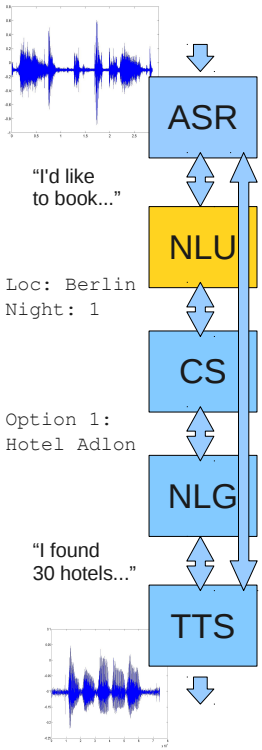
I'd like to book

→

Frame (incremental)

```
city:
arrival:
departure:
duration:
# of people:
room type:
```

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)

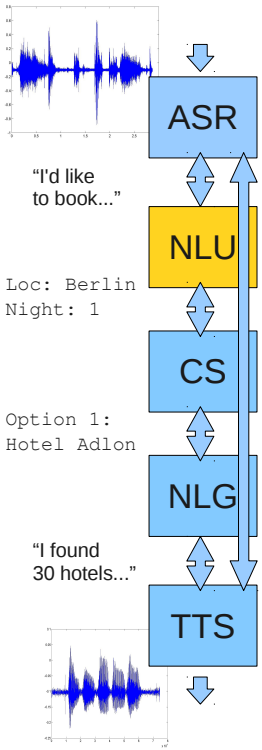
I'd like to book a double

→

Frame (incremental)

```
city:
arrival:
departure:
duration:
# of people:
room type:        double
```

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)

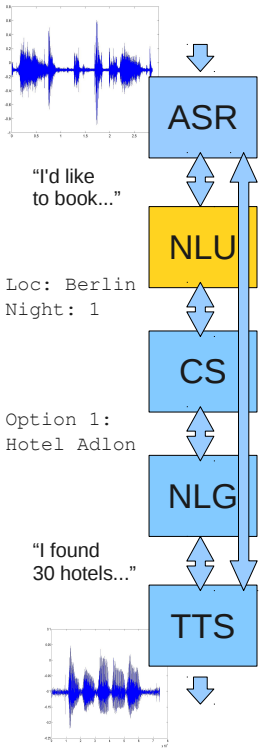
I'd like to book a double room for 2 adults

→

Frame (incremental)

```
city:
arrival:
departure:
duration:
# of people:        2
room type:        double
```

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)

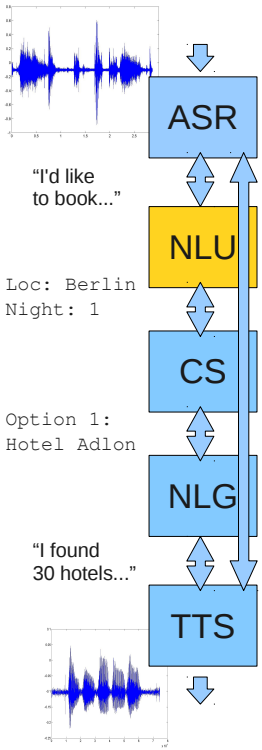
I'd like to book a double room for 2 adults in Berlin

→

Frame (incremental)

city:	Berlin
arrival:	
departure:	
duration:	
# of people:	2
room type:	double

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)

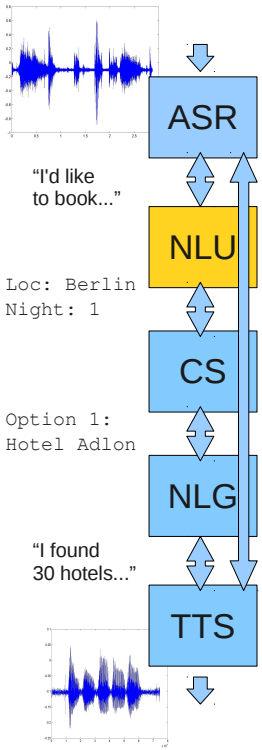
I'd like to book a double room for 2 adults in Berlin for one night.

→

Frame (incremental)

city:	Berlin
arrival:	
departure:	
duration:	1
# of people:	2
room type:	double

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)

I'd like to book a double room for 2 adults in Berlin oh, I meant Potsdam.

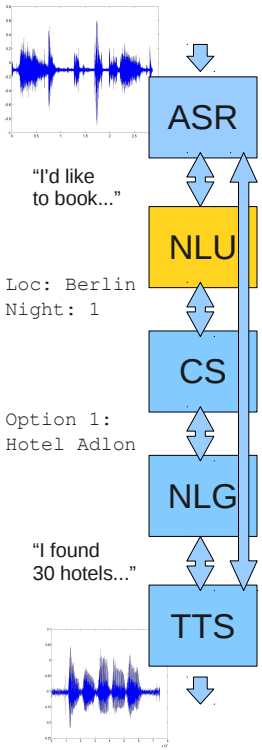
→

Frame (incremental)

city:	~~Berlin~~Potsdam
arrival:	
departure:	
duration:	1
# of people:	2
room type:	double

- Need to cope with self-corrections.

So how should an incremental-response NLU look?



Example for a hotel booking system

ASR output (incremental)

I'd like to book a double room for 2 adults in Berlin near Potsdam.

→

Frame (incremental)

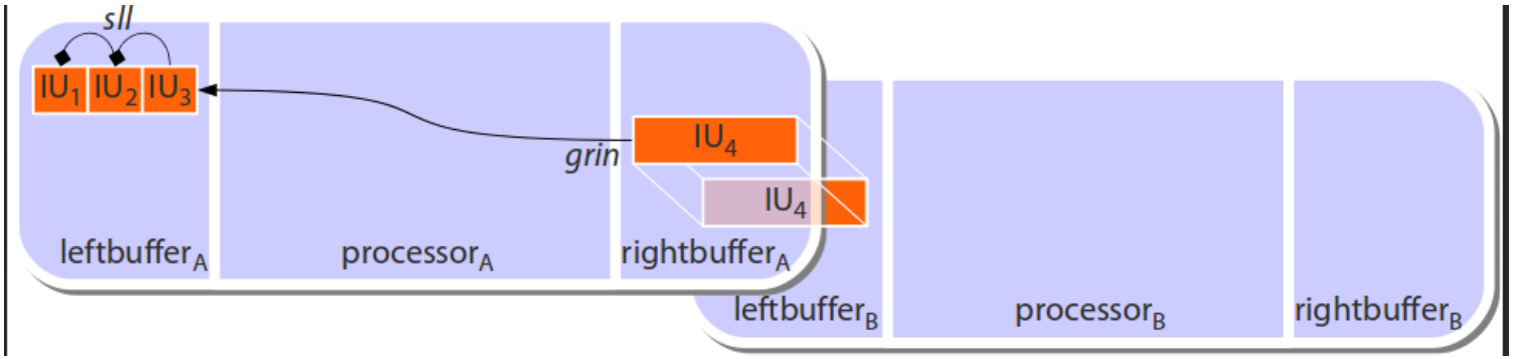
city:	Berlin
arrival:	
departure:	
duration:	1
# of people:	2
room type:	double

- Keyword-spotting vs. full parsing / semantic interpretation

Do predictive systems exist?

Baumann et al. (2010): InproTK toolkit.

- Divides semantics into "Incremental Units".
- Network of incremental units connected across processing window.
- Predicted IUs retracted or committed.



How do you predict when to commit something?

Depends on how. Baumann and Schlangen (2011), InproTK toolkit:

- Predictive models to complete user's current word.
- If you know what next word is, when will the previous word end?
- Tested three strategies:
  - Immediate speech – as soon as prediction occurs,
  - ASR prediction – use ASR model to generate lookahead to end of word.
  - TTS prediction – synthesize word using TTS, use simulation to connect phonemes.

Experimental evaluation against humans reading out loud: TTS prediction best.

But once you've done the interpretation. . .

. . . how do you generate the response?

- This is where we get to the CS and NLG portion of our show.
- We need to define the basic units of "output" content.
- Then need to relate them to output strings.

This is where we get into fine-grained questions of linguistic complexity.

- We're (so far) quantifying linguistic complexity with surprisal (but other candidates exist).
- How do we react to this quantification?

That's where statistical NLG comes in.

How did people generate sentences in the past?

- Natural language generation (NLG) rather late to the statistics party.
- Two kinds of previous attempts at including statistics:
  - Generate-and-filter (or rerank) from a handcrafted generator.
  - Embedding statistically-derived parameters into the generation model itself.

Should not surprise you that this is not optimal.

The real problem is that semantics is messy.

What we really want is a system that maps the semantics onto the output *dynamically*.

- Hence, BAGEL: "Bayesian networks for generation using active learning." (Mairesse et al. 2010)
- Goal: learn from *aligned data*. (So it looks a bit like machine translation.)
- Shift the emphasis from grammar-handcrafting to data alignment.

But we want to build complexity-aware NLG.

- We do not want to have to build the adaptation directly into our handcrafted grammar. (Too many decisions to make.)
- We have statistically-oriented parameters that permit word-by-word measurement of complexity.

For NLG, we need a syntax-semantics mapping.

This requires the notion of the "output" semantic unit. BAGEL defines the notion of a "semantic stack".

- Take a collection of underlying semantic concepts.
- Embed them into "stacks" of nested predicates.

inform(area(center))

- Outermost ("bottom") is usually a dialogue act: "inform"
- "Top" is the attribute being conveyed by the act. (I presume some flexibility in this representation.)

There are two kinds of semantic stacks.

- "Mandatory" (S_m): things that HAVE to appear in the output utterance.
- "Intermediary" (S_i): things that may be chosen to appear in the output representation (e.g. in order to make the utterance make sense).

This is what it looks like.

Mairesse et al. do some "crowdsourced" annotation for restaurant recommendations:

Charlie Chan	*is a*	*Chinese*	*restaurant*	*near*	*Cineworld*	*in the*	*centre of town*
Charlie Chan **name** **inform**	`inform`	**Chinese** **food** **inform**	**restaurant** **type** **inform**	`near` `inform`	**Cineworld** **near** **inform**	`area` `inform`	**centre** **area** **inform**
$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$	$t = 6$	$t = 7$	$t = 8$

Table 1: Example semantic stacks aligned with an utterance for the dialogue act `inform(name(Charlie Chan) type(restaurant) area(centre) food(Chinese) near(Cineworld))`. Mandatory stacks are in bold.

So now we can get our hands dirty.

The generation problem that BAGEL is solving.

- Given a length L, the number of realization phrases:
  - Find the most likely sequence of realization phrases $R* = (r_1 \ldots r_L)$ that
  - matches an unordered set of mandatory stacks S_m smaller than or equ al to L,
  - by deriving the optimum sequence of stacks $S*$ that may include intermediary stacks.

In other words, we want to maximize $P(R|S_m)$.

Q: How do we do this?

Here's where things get "mathy".

We need to estimate $P(R|S_m)$. We do this by "marginalizing" over all sequences in $Seq(S_m)$.

$$
\begin{aligned}
P(\mathbf{R}|\mathcal{S}_m) &= \sum_{\mathbf{S}\in Seq(\mathcal{S}_m)} P(\mathbf{R},\mathbf{S}|\mathcal{S}_m) \\
&= \sum_{\mathbf{S}\in Seq(\mathcal{S}_m)} P(\mathbf{R}|\mathbf{S},\mathcal{S}_m)P(\mathbf{S}|\mathcal{S}_m) \\
&= \sum_{\mathbf{S}\in Seq(\mathcal{S}_m)} P(\mathbf{R}|\mathbf{S})P(\mathbf{S}|\mathcal{S}_m) \qquad (1)
\end{aligned}
$$

Problem is, this is very expensive.

Q: How to deal with the expense?

Assume that the most optimal sequence produces the best realization.

$$P(\mathbf{R}|\mathcal{S}_m) \approx P(\mathbf{R}|\mathbf{S}^*)P(\mathbf{S}^*|\mathcal{S}_m) \quad (2)$$

$$\text{with } \mathbf{S}^* = \underset{\mathbf{S} \in \mathcal{S}eq(\mathcal{S}_m)}{\operatorname{argmax}} P(\mathbf{S}|\mathcal{S}_m) \quad (3)$$

But we need to figure out what the most optimal sequence is.

So we need a model that "decodes" the stack sequence:

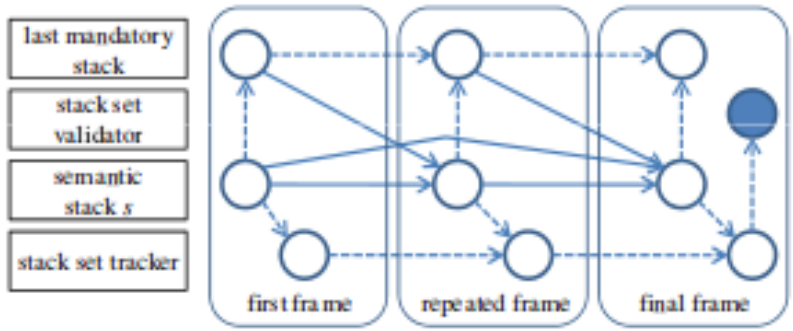


Figure 1: Graphical model for the semantic decoding phase. Plain arrows indicate smoothed probability distributions, dashed arrows indicate deterministic relations, and shaded nodes are observed. The generation of the end semantic stack symbol deterministically triggers the final frame.

Then we can find the best realization phrase...

...since we now have length $L = |S*|$.

$$\mathbf{R}^* = \underset{\mathbf{R}=(r_1 \ldots r_L)}{\operatorname{argmax}} P(\mathbf{R}|\mathbf{S}^*) P(\mathbf{S}^*|\mathcal{S}_m) \quad (4)$$

$$= \underset{\mathbf{R}=(r_1 \ldots r_L)}{\operatorname{argmax}} P(\mathbf{R}|\mathbf{S}^*) \quad (5)$$

And of course we need a model for that too.

A model for that too.

This is what actually maximizes $P(R|S*)$:

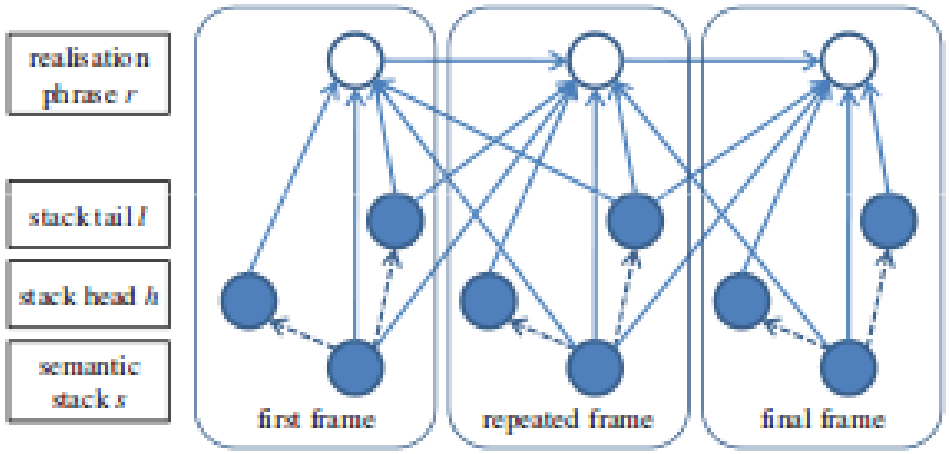


Figure 2: Graphical model for the realisation phase. Dashed arrows indicate deterministic relations, and shaded node are observed.

The problem is, what is dependent on what? Combinatorial problem.

But that all said. . .

. . . let me remind you again that this talk is somewhat aspirational.

- *Some* statistical model of realization is necessary for complexity-aware NLG.
- Once you *have* this model, you can figure out a way to penalize/boost some outputs vs. others.
- One advantage of Mairesse et al. for this: separate models for content and realization string.
  - Possible limited information presentation via penalties in the CS part.
  - Possible limited control of fine-grained syntactic/semantic presentation via the "output" NLG.

Maybe one of us here will figure it out :)

User adaptation, however, really does exist.

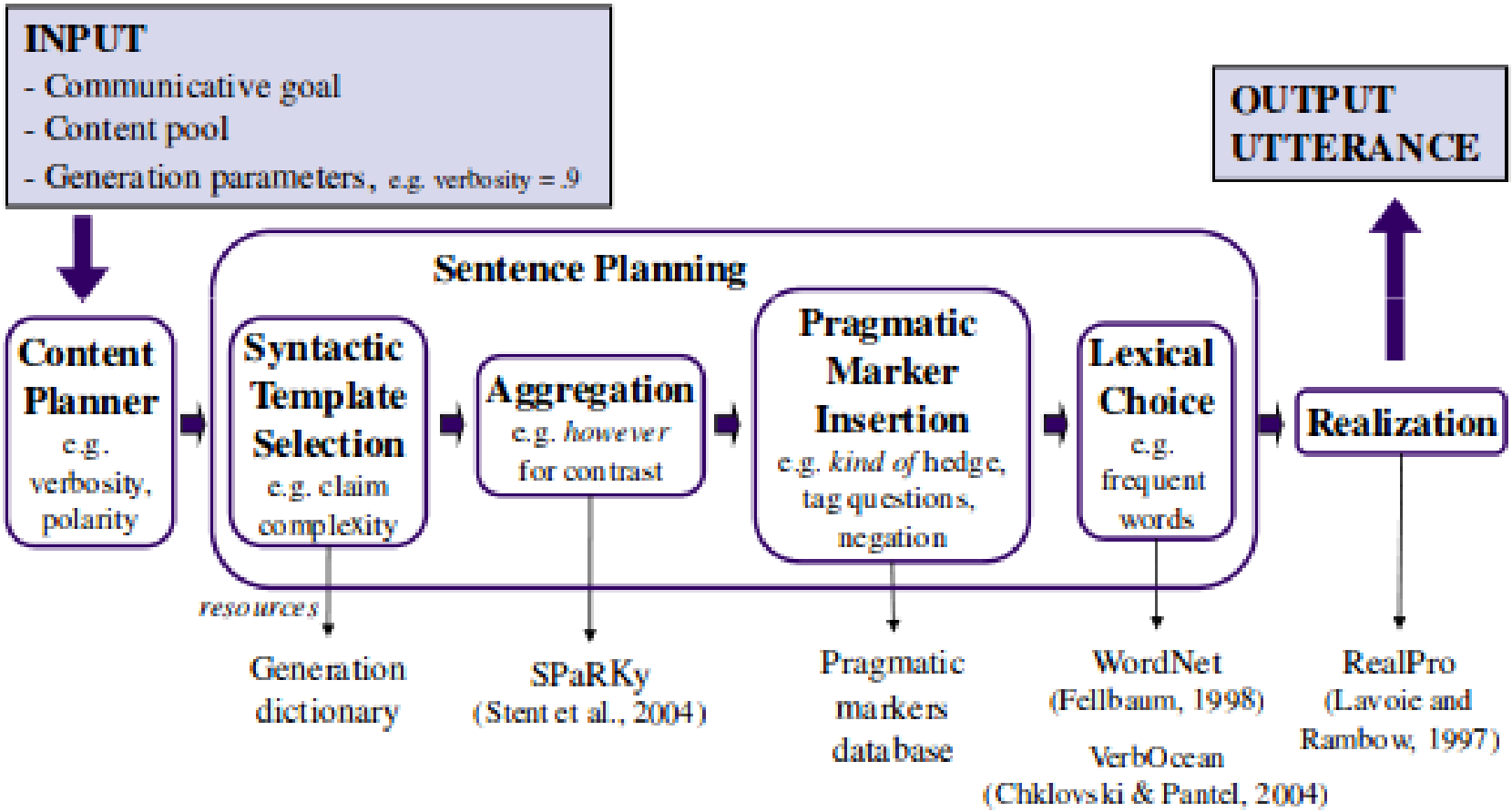
For example, personality-based adaptation. Mairesse and Walker (2010):

- Big 5 personality traits: Extraversion, emotional stability, agreeableness, conscientiousness, openness to experience.
- Can be associated with linguistic style: previous studies exist to correlate it to behaviour.
- People generate a lot of descriptive adjectives when talking to a "close friend" – can be used to generate personality features to train a model.

What kind of personality features are relevant?

Parameters	Description
VERBOSITY	Control the number of propositions in the utterance
RESTATEMENTS	Paraphrase an existing proposition, e.g. *'X has great Y, it has fantastic Z'*
REPETITIONS	Repeat an existing proposition
CONTENT POLARITY	Control the polarity of the propositions expressed, i.e. referring to negative or positive attributes
REPETITION POLARITY	Control the polarity of the restated propositions
CONCESSIONS	Emphasize one attribute over another, e.g. *'even if X has great Z, it has bad Y'*
CONCESSION POLARITY	Determine whether positive or negative attributes are emphasized
POLARIZATION	Control whether the expressed polarity is neutral or extreme
POSITIVE CONTENT FIRST	Determine whether positive propositions are uttered first
REQUEST CONFIRMATION	Begin the utterance with a confirmation of the request, e.g. *'did you say X?'*
INITIAL REJECTION	Begin the utterance with a rejection, e.g. *'I'm not sure'*
COMPETENCE MITIGATION	Express the speaker's negative appraisal of the hearer's request, e.g. *'everybody knows that ...'*

What does the overall model look like?



And some output rated for personality fit.

Trait	Set	Example output utterance	Score
Extraversion	low	Chimichurri Grill isn't as bad as the others.	1.00
	high	I am sure you would like Chimichurri Grill, you know. The food is kind of good, the food is tasty, it has nice servers, it's in Midtown West and it's a Latin American place. Its price is around 41 dollars, even if the atmosphere is poor.	6.33
Emotional stability	low	Chimichurri Grill is a Latin American restaurant, also it's located in Midtown West. It has quite friendly waiters. It offers adequate food. I imagine you would appreciate it.	2.92
	high	Let's see what we can find on Chimichurri Grill. Basically, it's the best.	6.00
Agreeableness	low	I mean, Chimichurri Grill isn't as bad as the others. Basically, the staff isn't nasty. Actually, its price is 41 dollars. It's damn costly.	2.00
	high	You want to know more about Chimichurri Grill? I guess you would like it buddy because this restaurant, which is in Midtown West, is a Latin American place with rather nice food and quite nice waiters, you know, okay?	5.75
Conscientiousness	low	I am not kind of sure pal. Err... Chimichurri Grill is the only place I would advise. It doesn't provide unfriendly service! This restaurant is damn expensive, its price is 41 dollars.	3.00
	high	Let's see what we can find on Chimichurri Grill. I guess you would like it since this eating house, which offers sort of satisfying food and quite satisfactory waiters, is a Latin American eating place.	6.00
Openness to experience	low	Err... I am not sure. Mmhm... I mean, Chimichurri Grill offers like, nice food, so I would advise it, also the atmosphere is bad and its price is 41 dollars.	3.50
	high	You want to know more about Chimichurri Grill? I believe you would love it, you know. I guess it's in Midtown West. Although this eating house's price is around 41 dollars, the food is rather satisfactory. This eating place, which provides kind of second-rate atmosphere, is a Latin American restaurant, alright?	5.00

But let's step back a bit.

We've focused really tightly on driving for a good reason.

- But there's more to dual-task activities than driving!
- Active areas of research, e.g.:
  - Disaster response.
  - Minimally-invasive surgery.

Let's talk about surgery!

Surgeons often interact with computer systems during procedures. Graetzel et al. (2004):

- Skeptical about dialogue systems in OR: too noisy.
  - (But there is active research in improving this since, e.g. Alapetite 2006.)
- Surgeons ask assistants to manipulate mouse for use with endoscopy systems, etc.
- Graetzel et al. are proposing a gesture-based system.

This is what spoken dialogue currently looks like in OR.

Humans as dialogue agent in Graetzel et al.

Move that mouse!

surgeon. Move the mouse to the third button down.
assistant. This one?
surgeon. No, the next one down.
assistant. This one?
surgeon. No, the other one... Yes, thats it.

If the surgeon must work while remote-controlling the mouse, it's a bit like driving.

Thing is, the constraint is a bit artificial.

Surgeon shouldn't have to demand mouse moves to get something done!

- Like every automation problem, should just have to input the intended goal.
- Then subject to all the workload issues we've discussed so far.

Computers with personality.

Well," the [elevator's] voice trickled on like honey on biscuits, theres the basement, the microfiles, the heating system . . .er. . ."

It paused. "Nothing particularly exciting," it admitted, "but they are alternatives."

"Holy Zarquon," muttered Zaphod, "did I ask for an existential elevator?" He beat his fists against the wall.

"Whats the matter with the thing?" he spat.

"It doesnt want to go up," said Marvin simply. "I think its afraid."

"Afraid?" cried Zaphod. "Of what? Heights? An elevator thats afraid of heights?"

"No," said the elevator miserably, "of the future."

"The future?" exclaimed Zaphod. "What does the wretched thing want, a pension plan?"

— *The restaurant at the end of the universe*, Douglas Adams.

So long, and thanks for all the fish!