Computational Semantics and Pragmatics

Autumn 2011

Raquel Fernández

Institute for Logic, Language & Computation
University of Amsterdam

Overview

Last week:

- BDI approaches vs. ISU approaches
  - ∗ ISU approach focuses on the public aspects of dialogue, on the common ground built by the participants during a conversation, and in how dialogue acts update the conversational scoreboard.
- Models of grounding
  - ∗ Dialogue as a joint action, requiring acts by speaker and addressee.
  - ∗ Grounding: the process whereby participants contribute to their common ground, giving feedback about their level of understanding.

Plan for today:

- Computational approaches to ISU and grounding in spoken dialogue systems

Grounding (recap)

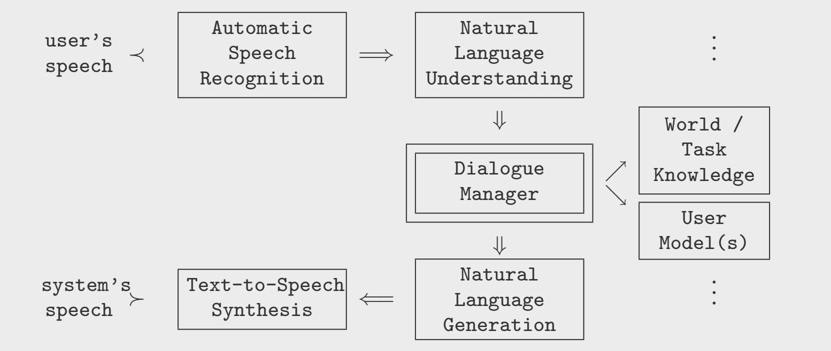
Level	Actions
1 contact:	A and B pay attention to each other
2 perception:	B perceives the signal produced by A
3 understanding:	A understands what B intends to convey
4 uptake:	B accepts / reacts to A's proposal

- effective utterances in dialogue are joint actions
- participants must understand each other at each level of communication up to the grounding criterion
- lack of understanding may occur at any level of action
- participants use different feedback mechanisms (positive and negative) to give evidence of understanding or lack thereof
- principle of least collaborative effort: participants will try to invest the minimum effort that allows them to reach the grounding criterion.

⇒ How are these ideas put into practice in dialogue systems?

Dialogue Systems Architecture

- Spoken Dialogue Systems (SDS) require an end-to-end architecture, where all sub-systems of language processing are at play (in different degrees of sophistication).
- The main components of an SDS are the following:



Grounding in Dialogue Systems

There are many factors that make dialogue systems prone to misunderstanding:

- limitations of automatic speech recognition
- limited linguistic knowledge (vocabulary, constructions,...)
- limited background knowledge

⇒ Error handling methods are therefore critical.

There are two main aspects involved in error handling:

- Error detection: has a communication problem occurred? (has the grounding criterion been reached?)
- Error repair: if a problem has been detected, how can it be signalled and repaired? (evidence of understanding and feedback)

Speech Recognition Problems

Most work on error handling focuses on error at the level of speech recognition. The typical *word error rate* is 20-30%.

S:	What city are you leaving from?
U:	Urbana Champaign [OKAY IN THAT SAME PAY]
S:	Sorry I'm not sure I understood what you said. What city are you leaving from?
U:	Urbana Champaign [FOR MINUTE SINCE HEY]
S:	I'm still having trouble understanding you. To go on, I need you to answer the following question. What city are you leaving from?
U:	Chicago [CHICAGO]
S:	Traveling from Chicago. Where would you like to go?
U:	Huntsville [SEOUL]
S:	Traveling to Seoul. What day did you need to travel?
U:	no no I'm traveling to Birmingham [THE TRAVELING TO BERLIN P M]
S:	Traveling in the afternoon. Okay what day would you be departing chicago
U:	/uh/ the tenth of august [AT THE TENTH OF AUGUST]
S:	A flight tomorrow. I have a Korean Airlines flight departing Chicago at 1:40pm, arrives Seoul at 5pm the next day. Do you want to take that?
U:	my destination is Birmingham [FLIGHT DESTINATION MR WEEKEND]
S:	Traveling on Saturday, August 12th. I have a flight departing Chicago at 1:40pm arrives Seoul at ...

Example interaction from Bohus (2007) *Error awareness and recovery in conversational spoken language interfaces*

Automatic Speech Recognition

The output of an ASR is an n-best list of hypotheses together with confidence scores indicating how confident the system is that each hypothesis corresponds to the utterance uttered by the interlocutor.

Rank	Path	AM logprob	LM logprob
1.	it's an area that's naturally sort of mysterious	-7193.53	-20.25
2.	that's an area that's naturally sort of mysterious	-7192.28	-21.11
3.	it's an area that's not really sort of mysterious	-7221.68	-18.91
4.	that scenario that's naturally sort of mysterious	-7189.19	-22.08
5.	there's an area that's naturally sort of mysterious	-7198.35	-21.34
6.	that's an area that's not really sort of mysterious	-7220.44	-19.77
7.	the scenario that's naturally sort of mysterious	-7205.42	-21.50
8.	so it's an area that's naturally sort of mysterious	-7195.92	-21.71
9.	that scenario that's not really sort of mysterious	-7217.34	-20.70
10.	there's an area that's not really sort of mysterious	-7226.51	-20.01

Confidence scores can be used to detect potential problems and to decide on a grounding strategy.

For details on the process of automatic speech recognition, see chapters 9 and 10 in Jurafsky & Martin (2009) *Speech and Language Processing*.

Exploiting ASR Confidence Scores

Let us assume the system takes into account only the top hypothesis together with its confidence score (h_1, σ_1).

An error detection mechanism can take the following form:

- set a confidence threshold θ ($\approx$ grounding criterion)
- if $\sigma_1 > \theta$, assume there is sufficient understanding
- if $\sigma_1 < \theta$, assume a problem has occurred

Different sub-thresholds ($\theta > \theta_a > \theta_b > \theta_c$...) can be used to trigger grounding actions for error repair. For instance:

- if $\theta > \sigma_1 > \theta_a$, use implicit verification (repetition or reformulation)
- if $\theta_a > \sigma_1 > \theta_b$, use explicit verification (clarification request – requires reaction)
- if $\theta_b > \sigma_1$, ask the interlocutor to repeat

Exploiting ASR Confidence Scores

Different sub-thresholds ($\theta > \theta_a > \theta_b > \theta_c ...$) can be used to trigger grounding actions for error repair. For instance:

- if $\theta > \sigma_1 > \theta_a$, use implicit verification (repetition or reformulation)
- if $\theta_a > \sigma_1 > \theta_b$, use explicit verification (clarification request – requires reaction)
- if $\theta_b > \sigma_1$, ask the interlocutor to repeat

S: Where would you like to go?
U: Huntsville [SEOUL]
S_1: Travelling to Seoul. What day did you need to travel? [*implicit verification*]
S_2: Did you say to Seoul? [*explicit verification*]
S_3: Sorry, I'm not sure I understood you. Where would you like to travel to?

Multiple Factors

Grounding strategies for error repair may take into account additional factors besides ASR confidence scores. For instance:

- number of previous clarification questions
- number of previous mis-recognitions
- the importance of a concept for a given task

Grounding actions may also be associated with *efficiency costs*

- defined, for instance, as the average length in syllables required to complete a repair sequence given a particular action
- this can be estimated from annotated dialogue corpora

$\Rightarrow$ The goal is to minimize collaborative effort (recall Clark's principle of least collaborative effort)

Data-driven Methods

Establishing the grounding status of an utterance (error detection) and deciding on a grounding action (error repair) can be treated as a classification problem.

- Supervised machine learning methods
  - classes: degrees of understanding/types of grounding actions
  - features: characteristics of the current dialogue states, dialogue history, cost of actions,...
  - need annotated dialogue corpora for training
- Implicit supervised methods
  - start with a uniform strategy (always explicit confirmation)
  - define/refine a confidence threshold model given the user's feedback

S: Where are you leaving from?
U: The airport.[LIBERTY AND WOOD]
S: Leaving from Liberty and Wood. Is that correct?
U: No.

Back to ISU

The ISU approach is influenced by two main traditions:

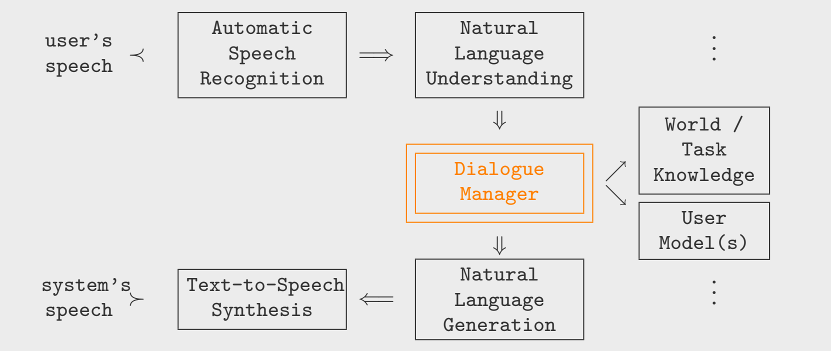
- The work of philosophers such as Lewis and Stalnaker
  - ∗ focus on public/conventional aspects of dialogue (*common ground*).
  - ∗ the dynamics of dialogue can be modelled using a game metaphor: participants (players) make moves that update an evolving *conversational scoreboard* that represents the information that has become common as a result of the dialogue.

Lewis. 1979. Score keeping in a language game. *Journal of Philosophical Logic*.
Stalnaker. 1979. Assertion. In *Syntax and Semantics IX*. Academic Press.
Carlson. 1983. Dialogue Games. *Synthese Language Library*. D. Reidel.

- The work of conversational analysts (Schegloff et al.) and psycholinguists (Clark et al.)
  - ∗ focus on interaction management and meta-communication
  - ∗ *grounding*

Allwood (1995) An activity-based approach to pragmatics. *Göteborg Papers of Theoretical Linguistics*.
Clark & Schaefer (1989) Contributing to discourse, *Cognitive Science*.
Schegloff et al. (1977) The preference for self-correction in the organization of repair in conversation, *Language*.

Dialogue Systems Architecture



Information State Update

According to the ISU framework, in abstract terms a dialogue can be modelled as:

- A set S of **dialogue states**, representing possible configurations of the conversational scoreboard;
- A set M of **dialogue acts**, which act as context-change operators;
- An **update function** $\delta : (S \times M) \rightarrow S$, that updates the conversational scoreboard given the current state of the dialogue and a new dialogue act.
- m is a **coherent next move** at a state s iff $\delta(s, m)$ is defined.

Several issues need to be worked out in detail, including:

- what information do dialogue states keep track of?
- what is the inventory is dialogue acts?
- what is the exact specification of the update function/update rules?
- what strategy can be used to choose a next dialogue move from a set of possible coherent next moves?

Information States

The term Information state (IS) refers to the state of the dialogue: the dialogue context that gets updated with each dialogue move.

Different theories of the dynamics of dialogue will represent ISs differently. Some common IS components are:

- the commitments of the dialogue participants
- a stack of questions under discussion (QUD)
- the latest move made in the dialogue
- grounded and ungrounded information

ISs are typically represented as feature structures. For instance:

$$\begin{bmatrix} \textsc{com} & \textit{Set of Propositions} \\ \textsc{qud} & \textit{Stack of QUDs} \\ \textsc{moves} & \textit{List of moves} \\ \textsc{pending} & \textit{List of moves} \end{bmatrix}$$

Traum & Larsson (2000) The Information State Approach to Dialogue Management. In *Current and New Directions in Discourse and Dialogue*, pp. 325–353.

Update Rules

Dialogue acts trigger IS updates. Update rules are specified in terms of:

- preconditions: information that must hold in the IS for the rule to be applied
- effects: the resulting IS after application of the rule

Dialogue acts can be described according to their IS update potential. For example:

- Questions add an element to QUD
- Answers eliminate an element in QUD
- Acknowledgements move information from PENDING to MOVES and COMMITMENTS
- ...

Dialogue Act Taxonomies

- Searle distinguishes between five basic types of speech acts: representatives, directives, commissives, expressives, declarations
- The DA taxonomies considered by ISU aim to cover a broader range of utterance functions and to be effective as tagsets for annotating actual dialogue corpora.
- Importantly, they include grounding-related acts.

- One of the most influential DA taxonomies is the DAMSL schema (Dialogue Act Markup using Several Layers) described by Core & Allen (1997).
  - *Forward Looking Functions* (FLF): initiating tags such as Assert, Info-Request and Offer that code how an utterance constrains the future dialogue.
  - *Backward Looking Functions* (BLF): reacting tags such as Answer, Accept, Reject, Completion, and Signal-non-Understanding that code how an utterance connects with the previous dialogue.

DA Taxonomies: SWBD DAMSL

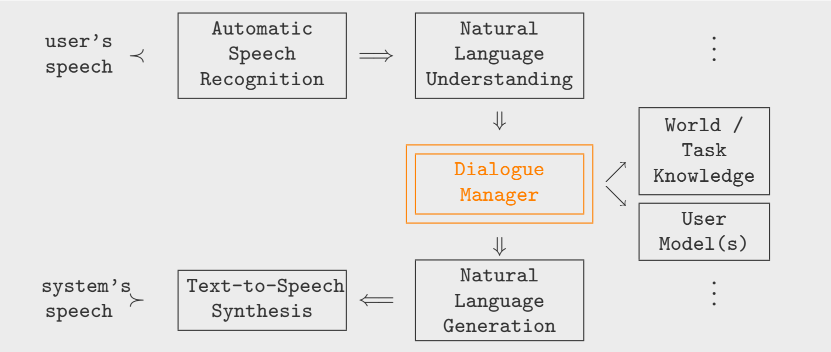
The SWBD DAMSL schema is a version of DAMSL created to annotated the Switchboard corpus. Here are the 18 most frequent DA in the corpus:

Tag	Example	Count	%
Statement	*Me, I'm in the legal department.*	72,824	36%
Continuer	*Uh-huh.*	37,096	19%
Opinion	*I think it's great*	25,197	13%
Agree/Accept	*That's exactly it.*	10,820	5%
Abandoned/Turn-Exit	*So, -/*	10,569	5%
Appreciation	*I can imagine.*	4,633	2%
Yes-No-Question	*Do you have to have any special training*	4,624	2%
Non-verbal	*<Laughter>,<Throat_clearing>*	3,548	2%
Yes answers	*Yes.*	2,934	1%
Conventional-closing	*Well, it's been nice talking to you.*	2,486	1%
Uninterpretable	*But, uh, yeah*	2,158	1%
Wh-Question	*Well, how old are you?*	1,911	1%
No answers	*No.*	1,340	1%
Response Ack	*Oh, okay.*	1,277	1%
Hedge	*I don't know if I'm making any sense*	1,182	1%
Declarative Question	*So you can afford to get a house?*	1,174	1%
Other	*Well give me a break, you know.*	1,074	1%
Backchannel-Question	*Is that right?*	1,019	1%

The average conversation consists of 144 turns, 271 utterances, and took 28 min. to annotate. The inter-annotator agreement was 84% (κ=.80).

http://www.stanford.edu/~jurafsky/manual.august1.html

Dialogue Management



The dialogue manager is the core component of a dialogue system:

- it keeps track of the dialogue context
- it integrates each incoming dialogue act into the context (the IS)
- it updates the state of the dialogue with a set of update rules
- it decides what to say next

What to say next

- A strategy for selecting particular actions at each state in a dialogue is called a policy (a mapping between states and actions).

  * *we have seen policies for grounding actions*
- Dialogue policies may be defined as a set of hand-crafted rules inspired by corpus data or may be derived statistically from data.
- Some stochastic approaches model the dialogue as a *Markov Decision Process* with *Reinforcement Learning*. Essentially:

  * A set S of states and a set A of actions
  * A set $F \subset S$ of preferred final states reflecting the overall goals of the system.
  * A transition probability $P_a(s, s')$ that indicates how probable it is that performing action a in state s will lead to state s'.
  * A reward function R, that assigns a reward to transitions $P_a(s, s')$ indicating how appropriate action a is at state s to achieve a state in F.

- The system is trained on many dialogues to learn a reward function that will allow it to find a policy that meets its goals.

Singh et al. (2002) Optimizing Dialogue Management with Reinforcement Learning: Experiments with the NJFun system, *Journal of Artificial Intelligence*, 16:105–133.

Next Week

Presentations of your final projects!

- slots of 17 minutes (12 + 5 minutes for questions)
- no more than 10 slides (send them to me beforehand)
- we will start sharp on time at 15h – please don't be late!
- feel free to invite fellow students to attend the workshop.