Dialogue 2

`stephen.pulman@comlab.ox.ac.uk`

Last time:

* we see dialogues as having a two level structure

* individual utterances express dialogue **moves**

* sequences of moves constitute a **game**.

Recognising what move(s) an utterance expresses tells you what the speaker wants or believes, and tells you what game(s) are being played, hence what to do or say next.

What is the correct set of move/game labels?

How do we recognise what move an utterance realises?

How do we deal with uncertainty?

A typical set of dialogue act labels:

```
* ASSERT ("This is my sister". Yes and no are also considered asserts,
      where they are responses to yes/no questions, thus abbreviated assertions.)
* OFFER ("Shall we look at another picture?")
* COMMIT ("Okay I'll do that")
* EXPRESSION (All social expressions such as "you're welcome".
              Also things like "wow!" and "great!")
* INFORMATION REQUEST (open question)
* CONFIRMATION REQUEST (yes/no question)
* REPEAT REQUEST ("Pardon?")
* ACTION DIRECTIVE ("Show me another one." All imperatives.)
* OPEN OPTION ("We could look at another picture."
           Stating an option in a way that doesn't demand an answer.)
* OPENING ("Hi")
* CLOSING ("Goodbye")
* ANSWER (An answer is invariably also an assert. Yes/no answers are asserts.)
* BACKCHANNEL ("Uhuh")
* REPEAT REPHRASE (Expressing understanding by paraphrasing)
* COMPLETION (Completing the utterance of the other speaker)
* NON-UNDERSTANDING ("I don't understand")
* CORRECTION (An assertion that corrects a previous assertion)
* ACCEPT (Accepting a proposal)
* REJECT (Rejecting a proposal)
```

Note that an utterance or section of an utterance may belong to several categories. For example, answers are also asserts though asserts may not be answers.

RECOGNISING DIALOGUE ACTS

We need to be able to:

- recognise which act(s) an utterance realises

- plan the next act

Factors involved:

- form of utterance (question, etc).

- intonation (Edwinstowe vs. Edwinstowe?)

- content: OK/Yes etc

- previous move(s): question-response, check-confirm

- beliefs about the other participants' beliefs (if speaker has been told P, but asks whether P, likely to be a check rather than a question)

How well can you do just by content?

1. Take move-annotated corpus (Map Task)

2. Collect samples of utterances for each type of dialogue move. Break into word sequences and count them. E.g. ‘Instruction’:

11 again	03 along
56 and	27 and then
01 and then at the	02 and then down
03 and then down south	....

3. Use a measure like ‘mutual information’ to find the most likely move for a new utterance, when broken into similar fragments.

$$\text{MI(Move,feature)} = \log \frac{\text{P(M,f)}}{\text{P(M)*P(f)}} = \log \frac{\text{P(f} \mid \text{M)}}{\text{P(f)}}$$

So given an utterance = $f_1 ... f_n$, and moves $M_1 ... M_m$, find the M_i that maximises:

$$\sum_{j=1...n} \log \frac{P(f_j | M_i)}{P(f_j)}$$

This is very simple. Does it work?

Correct answer in first N most probable:

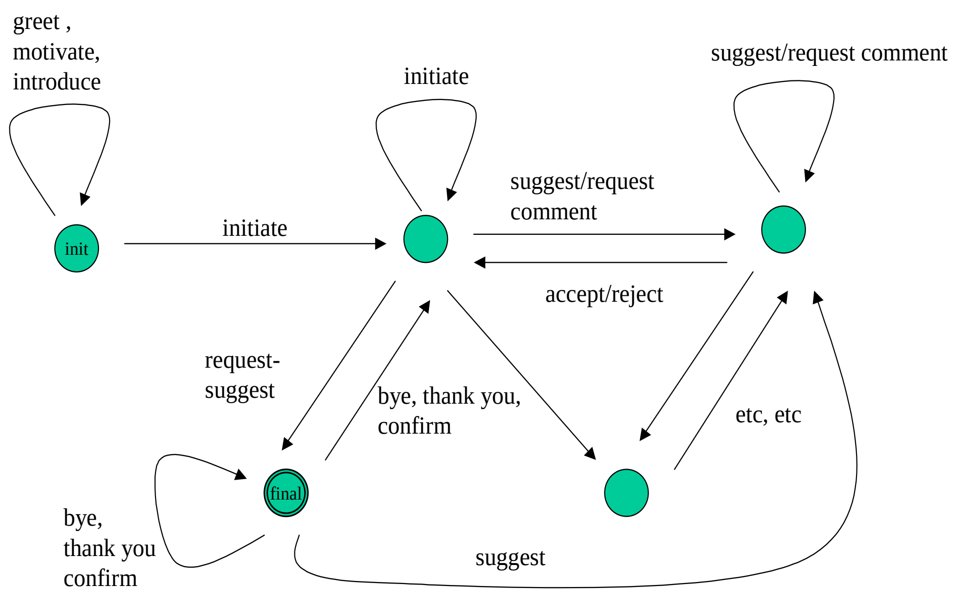
1/38% 2/57% 3/68% 4/72%

- wide variation for move types:

Acknowledge (72% correct)
Explain (30% correct).

Using previous moves: N-gram approach (e.g. Reitinger and Maier, Reitinger and Klesen)

1. Code moves as a finite state network, with probabilities attached to the transitions.



2. Train transition probabilities from an annotated corpus: P(Move$_n$ | Move$_{n-1}$, Move$_{n-2}$)

Achieves 69% accuracy, but Verbmobil dialogue acts are very task-specific. Can improve to 74% if word n-grams and speaker info is included.

On the ‘Switchboard’ corpus, Andreas Stolcke et al. (2000) (Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech, Computational Linguistics, Volume 26, Number 3)

`http://www.aclweb.org/anthology-new/J/J00/J00-3003.pdf`

used a HMM discourse act model, lexical information, cooccurrence, models of prosody, to get 65% accuracy on automatically recognised speech, 71% on human transcribed (human annotators get 84%)

Information State Update systems

Information state based: model mental state of dialogue participants as set of beliefs (+ desires and intentions)

> interpret utterance → update info state
> info state updates → plan new utterance

Example: WHQ: 'When is the next train?' would add to the current information state that: want(user,know(user,next-train))

This would trigger an update rule something like:

> IF want(user,know(user,X)) is in info state,
> THEN find Y = value-for(X),
> AND produce-answer(X is Y)

These rules are typically hand-coded rather than learned.

Dealing with uncertainty

- What did the user say?
- What was his/her intention?
- What should the system say next?
- How do we measure the ‘success’ of a dialogue?
- Recent approaches to planning use probabilistic reasoning.
- ...and a notion of ‘utilities’ of actions.
- Aim to maximise the utility of a dialogue?

Markov Decision Processes

Recent approaches to dialogue have explored MDPs, as a way of trying to ensure that a system behaves sensibly at each stage. Formally, MDP are characterized by the tuple S, A, T, R, where:

- S is the state space which describes the agent's world.
- A is the action space, actions that the agent may take.
- T defines the transition probability that action $a \in A$ in state $s \in S$ at time t will lead to state $s' \in S$ at time $t+1$, $\Pr(s_{t+1} = s' | s_t = s, a_t = a)$. (Estimated from a dialogue corpus)
- R defines the immediate reward for performing action a in state s, $r(s, a)$.

The goal of the agent is to choose actions which maximise its overall cumulative reward. An optimal policy can be precomputed saying for each state what the best action is. However, this is usually intractable.

Reinforcement learning

The method usually used to learn a reward function from initial settings is Q-learning, a form of reinforcement learning (i.e. from experience via punishments and rewards).The value $Q(s,a)$ is the expected discounted sum of future rewards obtained by taking action a from state s and following an optimal policy. The Q-values are learned from agent experience following these steps:

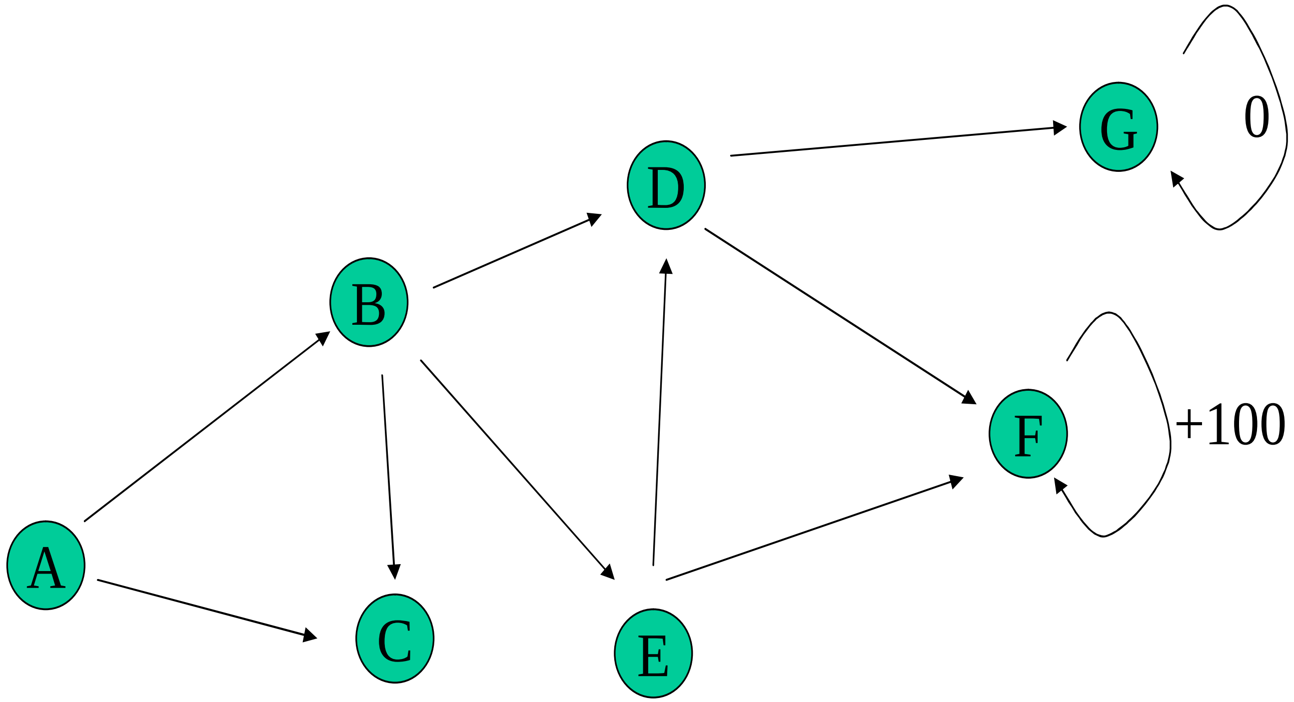
1. Pick a state s at random.

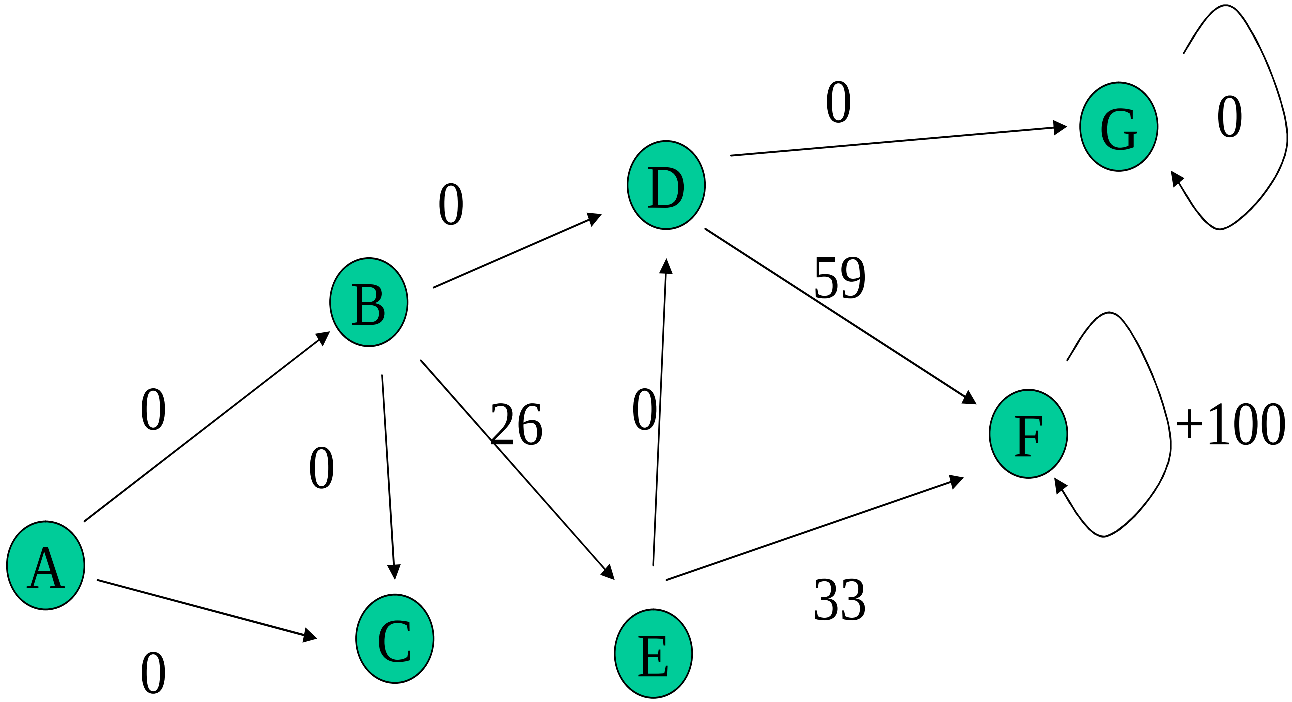
2. From the current state s, select an action a. This will cause a receipt of an immediate reward r, and arrival at a next state s'

3. Update $Q(s,a)$ based on this experience as follows:
$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_{b \in A} Q(s',b) - Q(s,a)] \tag{1}$$
where α is the learning rate and $0 < \gamma < 1$ is a 'discount factor'

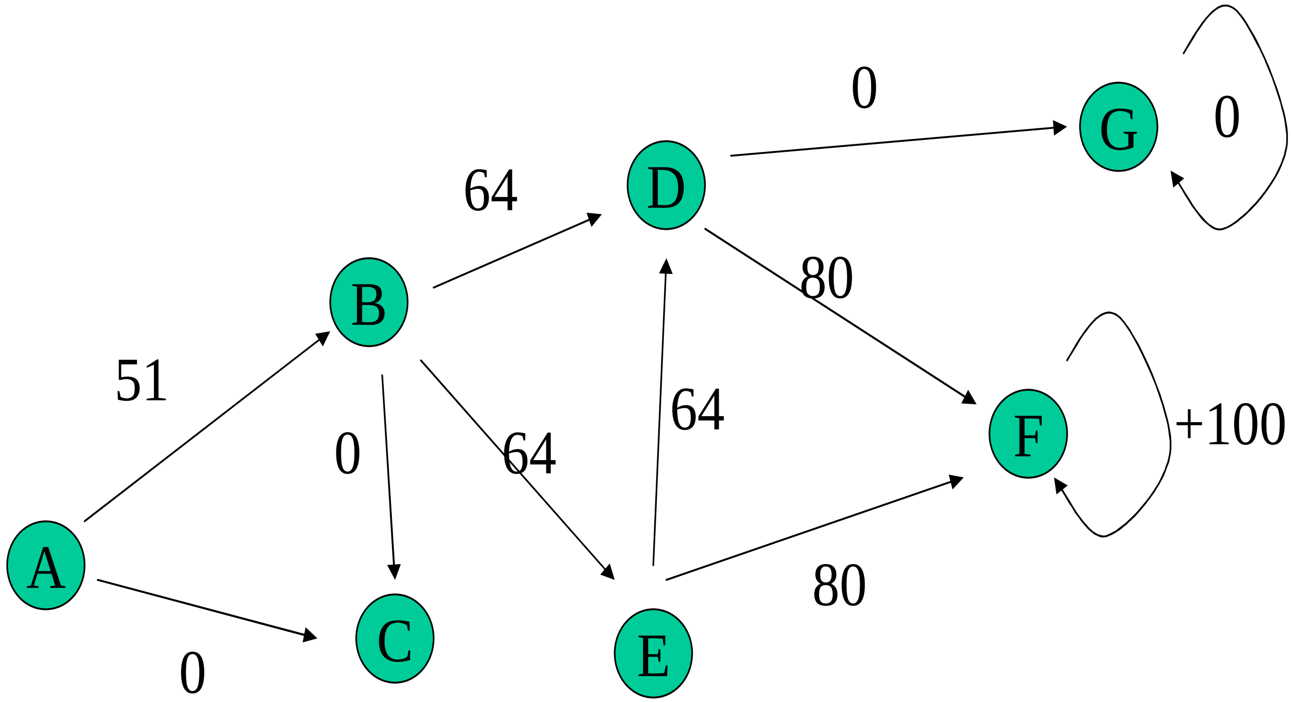
4. Go to 1

This algorithm is guaranteed to converge.


G
0
D
B
F
+100
A
C
E



after 5 iterations



at convergence

Partially Observable Markov Decision Processes

Because speech recognition and language understanding is not perfect, a system does not always know which state it is in. A POMDP maintains a probability distribution over states, $b(S)$, and adds an 'observation space' O with an associated distribution Z, which defines the probability of observing o' in state s' at time $t+1$ when the action a has been taken at time t, $\Pr(o_{t+1} = o' \mid s_{t+1} = s', a_t = a)$.

A POMDP, given the appropriate policy, operates as follows:

1. In $b(s)$, select an action a according to a policy
2. Get a reward $r(s, a)$
3. Go to state s', according to T
4. Make an observation from the environment according to $\Pr(o'|s', a)$
5. Update $b(s')$ (some complexity concealed here!)
6. Return to step 1

The various components of a POMDP correspond to dialogue system components as follows:

The belief space encodes information about what has been said, what the user's goals are, etc.

Actions are the range of things the system can say.

Observations are hypotheses about what the user has said. The associated probability distribution represents the certainty that the recogniser/NLU system has got things right

The transition function is essentially the dialogue act model, and the distribution can be estimated given annotated corpus data.

The reward function has to be learned from an initial setting: high values for successful end states, negative values for repeated clarifications, etc.

POMDP approaches have been shown to perform better (e.g. in terms of task completion) in domains like flight booking, ordering pizza, etc. than information state or MDP models.

This seems to be because they are less sensitive to recognition or NLP errors, carrying along multiple hypotheses via $b(S)$.

But they do not scale up well: the state space can get very large, so updating $b(S)$ or learning a policy becomes intractable.

For an amusing take on the current capabilities of spoken language dialogue systems see this:

`http://www.vuidesign.net/julie-meets-antonio-banderas-operator-video.htm`

Allen, J. and Core, M. *DAMSL: Dialogue Markup in Several Layers*, draft contribution for the Discourse Resource Initiative, 1997.

H. Bunt. Dynamic interpretation and dialogue theory. In M.M. Taylor, D.G. Bouwhuis, and F. Neel, editors, *The structure of multi-modal dialogue*, volume 2. Amsterdam John Benjamins, 1997.

Cooper, R. and Larsson, S. and Matheson, C. and Poesio, M. and Traum, D. *Coding Instructional Dialogue for Information States*, Trindi project deliverable D1.1. February 1999.

Cooper, R. and Larsson, S. *Dialogue Moves and Information States*, Technical Report, University of Gothenberg, 1998.

Ginzburg, J. *An update semantics for dialogue* in H. Bunt, R. Muyskens, G. Rentier (eds). Proceedings of the First International Workshop on Computational Semantics, Tilburg, 1994.

Ginzburg, J. *Interrogatives: Questions, Facts, and Dialogue*, in Handbook of Contemporary Semantic Theory, ed. S. Lappin, Oxfor: Blackwell Publishers Ltd. 385-422.

Kowtko, J. and Isard, S. and Doherty, G. M. *Conversational Games within Dialogue*, Research Paper HCRC/RP-31, Human Communication Research Centre, Edinburgh University.

Poesio, M. and Traum, D. 1997, *Conversational Actions and Discourse Situations*, Computational Intelligence 13(3), 309-347.

Poesio, M. and Traum, D. *Towards an Axiomatization of Dialogue Acts*. in TWLT 13, Formal Semantics and Pragmatics of Dialogue, ed. J. Hulstijn and A. Nijholt, Proceedings of the 13th Twente Workshop on Language Technology, Twente, Holland, 1998.

S. Bird et al 1995 Dialogue Move recognition using Topic Spotting techniques; ESCA Workshop on Spoken Dialogue Systems

K. Samuel et al 1998 Dialogue Act Tagging with Transformation Based Learning, in Proceedings of the 17th International Conference on Computational Linguistics (COLING-ACL '98)

N. Reithinger and E. Maier. Utilizing statistical dialogue act processing in Verbmobil. In *Proceedings of 33rd ACL, Cambridge Mass.*, pages 116–121, 1996.

J. Williams and S. Young (2007). "Partially Observable Markov Decision Processes for Spoken Dialog Systems." Computer Speech and Language 21(2): 231-422

Many other POMDP related papers at `http://mi.eng.cam.ac.uk/~sjy/publications.html`