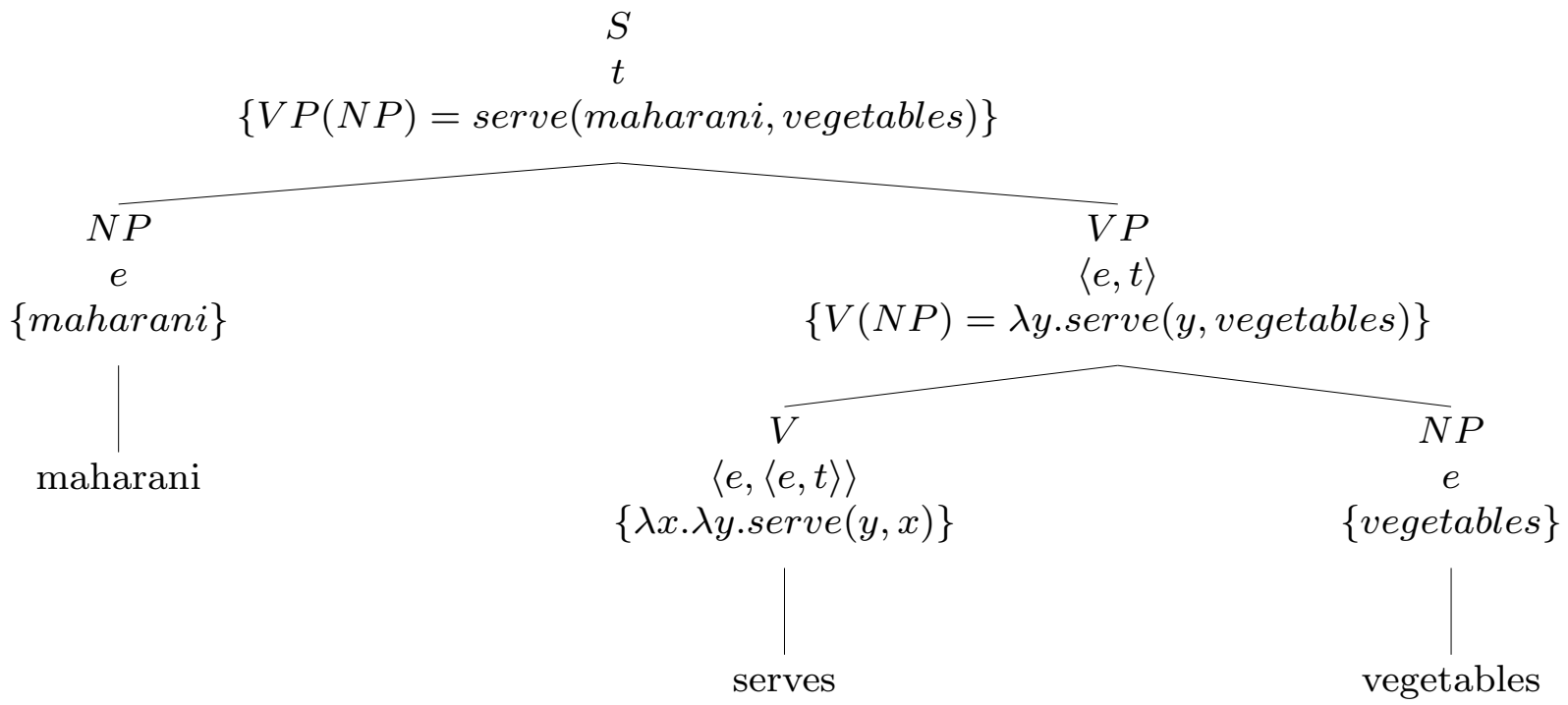
ECS763P
Natural Language Processing

Week 9
Discourse & Dialogue

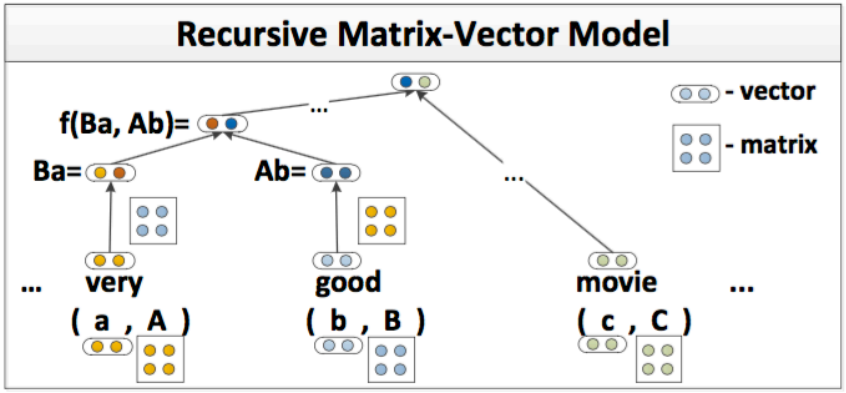
Matthew Purver
with material from Jurafsky & Martin:
Chapter 21 (discourse), 24 (dialogue)

Last week: Semantic Parsing



Logic + Statistics

- Practical semantic parsers will:
  - Use a different grammar (e.g. dependency parsing)
  - Use a different LF system (e.g. frames, DRT)
- And very importantly:
  - Use probabilistic parsing and/or classification to disambiguate
    - Now can be based on semantic features too!
- Maybe all three (e.g. neural nets, Socher et al 2012)
  - (but same basic insight)



- **<u>This is a very common pattern in building NLP applications:</u>**
  - **Logical framework to define task structure, impose constraints**
  - **Statistical (probabilistic, classifier etc.) model to disambiguate**

Why? Ambiguity!

- Parsing
  - Massive ambiguity
  - Perceived ambiguity ≠ potential ambiguity
  - i.e. humans don't notice the vast majority of ambiguities
    - "I'm interested in growing plants"
    - "Show me the meals on the flights from Phoenix"
      - 11 readings
  - Many WSJ sentences get **thousands** of parses
    - "Fed raises interest rates 0.5% in effort to control inflation"
      - Minimal grammar: 36 parses
      - Simple 10 rule grammar: 592 parses
      - Real-size broad-coverage grammar: millions of parses

- Semantic ambiguity
  - E.g. quantifier scope
    - "Everybody has a dream"

- And there are more sources of ambiguity to come
  - (wait)

Evaluation

- We haven't said much about evaluation recently …
- Text classification, sentiment analysis
  - Accuracy
  - But with uneven numbers/importance of classes: precision, recall, F-score (why?)
- Sequence labelling
  - As above, but be careful with units (tokens? sentences?)
  - Do some mistakes matter more than others?
- Topic modelling
  - Tricky! Human ratings?
- Parsing
  - Coverage (how many sentences could it parse?)
  - P/R/F of bracketed phrases (subtrees) in (N-)top-rated parse hypotheses
  - Error (e.g. crossing brackets)
- Semantic parsing
  - P/R/F of semantic triples
- Intrinsic vs extrinsic evaluation
  - The metrics above are all **intrinsic**
  - **Extrinsic**: measure the effect on the overall downstream task (IE, QA etc)

Information Extraction

- Turn unstructured text into structured information
  - Usually to populate a database
  - Usually given a set of standard templates
- This means we're looking for entities & events, and relations between them
  - i.e. semantic parser output!
  - But first we need to identify the entities & events themselves ...

- Pipeline of standard sub-tasks:
  - Named Entity Recognition
  - Temporal Expression Normalisation
  - Event Extraction
  - Relation Detection & Extraction
  - Template-Filling

  - *(i.e. logic + statistics!)*

Citing high fuel prices, [$_{ORG}$ **United Airlines**] said [$_{TIME}$ **Friday**] it has increased fares by [$_{MONEY}$ **$6**] per round trip on flights to some cities also served by lower-cost carriers. [$_{ORG}$ **American Airlines**], a unit of [$_{ORG}$ **AMR Corp.**], immediately matched the move, spokesman [$_{PER}$ **Tim Wagner**] said. [$_{ORG}$ **United**], a unit of [$_{ORG}$ **UAL Corp.**], said the increase took effect [$_{TIME}$ **Thursday**] and applies to most routes where it competes against discount carriers, such as [$_{LOC}$ **Chicago**] to [$_{LOC}$ **Dallas**] and [$_{LOC}$ **Denver**] to [$_{LOC}$ **San Francisco**].

Question-Answering

- Providing answers to questions (usually **factoid questions**):
  - What is the capital of Estonia?
  - Who founded Virgin Airlines?
  - Which is the biggest port on the Mediterranean?
  - When is the next leap year?
  - **Answer Type:** usually person, location, date, city etc.
- Compare:
  - Information Retrieval: returning relevant text passages
  - Information Extraction: finding pre-specified information

- Two families of approaches:
  - IR ("text-based") method
    - Retrieve relevant short passages; rank by answer type
  - Semantic ("knowledge-based") method
    - Parse question & answer, match semantic relations

IR-based QA

- (You may have done this in the IR module)

- Question processing:
  - Extract keywords
    - (see IR module)
    - Maybe remove wh-words, do query expansion
  - Classify expected answer type
    - e.g. supervised classification, n-gram features

- Answer retrieval:
  - Document retrieval: standard IR
  - Passage retrieval: more specific passage
    - Standard IR methods, weighted keyword relevance
    - Rank based on number & type of named entities
  - Answer extraction
    - Template/pattern-matching
    - N-gram extraction & combination ("tiling")

Semantic QA

- Parse query: "which states border Texas?" $\lambda x.state(x) \wedge border(x, texas)$
- Parse **lots** of text into a knowledge base
- Similar basic processing steps to IE:
  - Named Entity Recognition (after POS-tagging etc)
  - Temporal Expression Normalisation
  - Semantic Parsing
- Query & candidate fact processing:
  - Parse to semantic LF for question
    - Often needs specific grammars, with supervised machine learning for disambiguation
  - Map LF to database/triple format
    - Rule-based (manually specify patterns)
    - Supervised learning: associate parse sub-trees with triples
    - Distant supervision
      - e.g. find common patterns "X borders Y"
      - Look for "X … Y" and find new common patterns "X is bounded by Y", "X neighbours Y" etc
  - *(i.e. logic + statistics!)*

$state(oklahoma)$

$border(oklahoma, texas)$

$state(kansas)$

…

What about …

- QA needs to use data like Wikipedia:

Richard Branson

From Wikipedia, the free encyclopedia

Sir Richard Charles Nicholas Branson (born 18 July 1950) is an English business magnate, investor and philanthropist.[4] He founded the Virgin Group, which controls more than 400 companies.[5]

Branson expressed his desire to become an entrepreneur at a young age. At the age of sixteen his first business venture was a magazine called *Student*.[6] In 1970, he set up a mail-order record business. In 1972, he opened a chain of record stores, Virgin Records, later known as Virgin Megastores. Branson's Virgin brand grew rapidly during the 1980s, as he set up Virgin Atlantic airline and expanded the Virgin Records music label.

- Who founded the Virgin Group?
- When did Richard Branson set up Virgin Atlantic?
- What was Branson's first business venture?

And what about …

- Did Branson enter the Australian aviation market? When?

The Sydney Morning Herald | Business

News & Views | Markets | Quotes | Workplace | Consumer | Property | Innovation

The year was 2001, one year after the legendarily brash, anti-establishment renegade had pinpointed an opportunity to enter the Australian aviation market. Branson did so against the backdrop of a corporate landscape littered with the carcasses of other airline aspirants such as Compass and Impulse, all of whom had tried and failed to get a foothold in a duopoly market locked up the then government-owned Qantas and the Singapore Airlines-Air New Zealand-controlled Ansett.

Semantics vs Pragmatics

- Pragmatics: meaning *beyond the sentence*
- **Context-sensitivity** of meaning
  - (not to be confused with *"context-sensitive grammars"*)
- In particular:
  - **Anaphora** (including temporal)
    - "He sold them to himself two days after that."
  - **Co-reference**
    - "The legendarily brash, anti-establishment renegade …"
  - **Ellipsis**
    - "She promised she would; but she didn't."
- Very important in semantic tasks (e.g. IE, QA)
  - Less important for others (e.g. document/sentiment classification) – why?
- More generally: **implicature & intention**
    - Could you pass the salt? Do you have the time?
    - I'm out of petrol. There's a garage round the corner
    - I employed Mr Smith from 2016 to 2017. His handwriting is very neat.
  - (But we're not going to worry about this … until next week)

Referring Expressions

- Five types of **referring expression (RE)**:
  - **Indefinite noun phrases ("a N", "some N")** introduce *new* referents
    - "I saw a nice car today. Some other people noticed it too"
  - **Definite noun phrases ("the N")** refer to *identifiable* referents
    - Sometimes identifiable from previous mention
      - "I saw a nice car today. I'd like to buy the car tomorrow."
    - Sometimes identifiable from world knowledge or the description itself:
      - "The Prime Minister is coming to tea."
      - "The car of the year is the Ford Dustpan."
  - **Pronouns ("she", "it")** refer to highly *salient* referents
    - "I saw a nice car today. I'd like to buy it tomorrow."
    - "She's coming to tea."
  - **Demonstratives ("this (N)", "those")** refer to *near* or *distant* referents (literally or metaphorically)
    - "This new car is much faster than that old one."
  - **Names** can refer to *old or new* referents
    - "The Prime Minister will visit President Trump today, Mrs May's office announced."

Pronouns (Anaphora)

- Pronouns refer to (salient) contextual elements:
- **Anaphora**: referring back to items already mentioned
  - (**antecedents**: previous referring expressions)
  - "Sue left her coat behind"
- **Cataphora**: referring forward to items to be mentioned
  - "Before she leaves, Sue always checks her coat"
- **Deixis**: referring to the current environment/situation
  - "I want you to leave now"
- **Discourse** and **temporal** reference:
  - "We'll be ready then. Well, that's good."
- **Bound** variables:
  - "Every dog has its day." ~~"It lives next door."~~

- Except **pleonastic** or **generic** uses, which do not refer:
  - "It's raining. Always happens when you least expect it."

Pronoun Resolution (LFs)

- For referential (non-generic) uses, we must identify (**resolve**) the antecedent
- We need a formalism which makes this possible
  - FOL doesn't do very well:
    - "Mary had a dog. It barked."

$$\exists x.dog(x) \wedge have(M,x)$$
$$bark(x)???$$

  - Discourse Representation Theory (DRT):

m,x
have(m,x)
dog(x)

m,x	
have(m,x)	bark(x)
dog(x)	

  - Allows merging of DRSs, subject to logical conditions:
    - Mary didn't have a dog. ~~It barked.~~"

m
¬ [have(m,x) ; dog(x)]

m	
¬ [have(m,x) ; dog(x)]	bark(?)

Pronoun Resolution: Constraints

- Huge potential ambiguity: most previous REs are candidate antecedents
  - Resolution is largely about resolving this ambiguity – how?

- Many hard constraints on possible antecedents:
  - **Number**
    - (English: singular vs plural)
    - "I have a dog and two cats. It is/They are very fluffy."
  - **Person**
    - (English: 1st, 2nd, 3rd)
    - "I saw you with John. You were / He was happy."
  - **Gender**
    - (English: male, female, nonpersonal)
    - "Sue met John and his dog. She/he/it was happy."
  - **Binding**
    - (English: reflexives with clause subjects)
    - "John thinks Bill likes him/himself."
- (We often don't even notice these potential ambiguities, as humans!)

Pronoun Resolution: Preferences

- And many softer preferences:
  - **Recency**
    - (more recent > less recent)
    - "Sue lives in Reading. Jane lives in Havant. She has a dog."
  - **Grammatical role**
    - (subject > object > other)
    - "Sue knows Jane. She is coming today."
  - **Repetition**
    - (more mentions > fewer mentions)
    - "Sue is a banker. She works in the City. Jane likes her. She drives a Jaguar."
  - **Parallelism**
    - "Sue has known Jane since 1989. Gretel has known her since 1982."
  - **Discourse / event semantics**
    - "Sue is annoyed with Jane. She spilt her drink."
    - "Sue loves her dog. We took her for a walk."

- (We are more likely to notice these ambiguities)

Pronoun Resolution

- So we can use these constraints in resolution
- Rule-based:
  - e.g. using Centering Theory (Grosz et al, 1995 – see Jurafsky & Martin)
  - Order REs and possible antecedent REs by prominence
    - e.g. subject > object > other
    - "Forward-looking centres" (FLCs) = all REs in sentence
    - "Backward-looking centres" (BLCs) = all REs mentioned in previous sentence
  - Filter possible pairings using hard constraints
  - Order possible pairings by transition rules
    - e.g. (new FLC = new BLC = old BLC) > (new FLC = new BLC ≠ old BLC) > (…)
- Statistical classification:
  - Supervised classification
  - Potential pronoun-antecedent pairs as instances
  - Features chosen to relate to constraints/preferences:
    - number, gender, person match
    - word/sentence/syntactic distance
    - grammatical/semantic role & parallelism
- (i.e. logic + statistics!)

Co-Reference Resolution

- More general version of the problem
  - Build **chains** of all co-referring expressions

Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 2004, saw her pay jump 20% to $1.3 million, as the 37-year-old also became the Denver-based financial services company's president. It has been ten years since she came to Megabucks from rival Lotsabucks.

Co-Reference Resolution

- More general version of the problem
  - Build **chains** of all co-referring expressions

Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 2004, saw her pay jump 20% to $1.3 million, as the 37-year-old also became the Denver-based financial services company's president. It has been ten years since she came to Megabucks from rival Lotsabucks.

Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 2004, saw her pay jump 20% to $1.3 million, as the 37-year-old also became the Denver-based financial services company's president. It has been ten years since she came to Megabucks from rival Lotsabucks.

Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 2004, saw her pay jump 20% to $1.3 million, as the 37-year-old also became the Denver-based financial services company's president. ~~It~~ has been ten years since she came to Megabucks from rival Lotsabucks.

Co-Reference Resolution

- Candidate REs identified by POS-tagging & parsing
  - Need to identify links i.e. pair REs with antecedents
- Approach similar to pronoun resolution:
  - But we now need to add other RE types
  - Particularly definite noun phrases
- Supervised classification:
  - (same approach as for pronoun resolution)
  - Pairs of correct/incorrect examples
  - Features:
    - Features used for pronoun resolution, plus:
    - Lexical similarity (e.g. RE/antecedent edit distance)
    - Semantic co-reference (e.g. dates)
    - Syntactic relation (e.g. apposition ("X, the Y, …")
    - POS/lexical type of head noun
  - Big search space: use e.g. graph-based methods
    - Efficient search, while finding optimal chains (not just independent pairs)

Verb Phrase Ellipsis

- "Branson **did so**"
- Similar problem, but with some important differences
- Identifying ellipsis sites is harder:
  - "I can't today but I will try tomorrow"
  - "I can't today but I will – try tomorrow"
  - Use lexical, POS sequence, syntactic tree features
- Resolving ellipsis is more complex:
  - Not a purely lexical or syntactic operation:
    - "John likes tennis. So do you."
  - But what's the semantic operation?

$$like(john, tennis) \quad \text{-->} \quad like(you, tennis)$$

  - Can use lambda calculus to generate possible antecedents
    - abstract subject:

$$like(john, tennis) \quad \text{-->} \quad \lambda x.like(x, tennis)$$

  - Resolve VPE sites to semantic antecedent functions

$$P(you) \quad \text{-->} \quad \text{P}=\lambda x.like(x, tennis) \quad \text{-->} \quad like(you, tennis)$$

Ellipsis & Ambiguity

- Unfortunately, this brings in ambiguity again:
  - Sometimes that's appropriate:
  - "Sue thinks John likes Jane. So does Bill."

$$think(sue, like(john, jane)) \text{ --> } \lambda x.think(x, like(john, jane))$$

$$think(sue, like(john, jane)) \text{ --> } \lambda x.like(x, jane)$$

  - But sometimes some are spurious:
  - "John likes his teacher and his classmates. Sue does too."

$$like(john, teacher(john) \wedge mates(john)) \text{ --> } \lambda x.like(x, teacher(x) \wedge mates(x))$$

$$like(john, teacher(john) \wedge mates(john)) \text{ --> } \lambda x.like(x, teacher(john) \wedge mates(john))$$

$$like(john, teacher(john) \wedge mates(john)) \text{ --> } \lambda x.like(x, teacher(x) \wedge mates(john))$$

$$like(john, teacher(john) \wedge mates(john)) \text{ --> } \lambda x.like(x, teacher(john) \wedge mates(x))$$

- Need ambiguity resolution e.g. machine learning using suitable features:
  - Parallelism (e.g. subject-subject)
  - Discourse coherence (e.g. from Rhetorical Structure Theory)
  - Semantic plausibility & event/role restrictions

Discourse Structure

- We can assign structure to whole texts
  - Role relations between clauses/sentences
  - "Discourse parsing" (e.g. Marcu, 2012)
    - e.g. Rhetorical Structure Theory (Mann & Thompson, 1993)

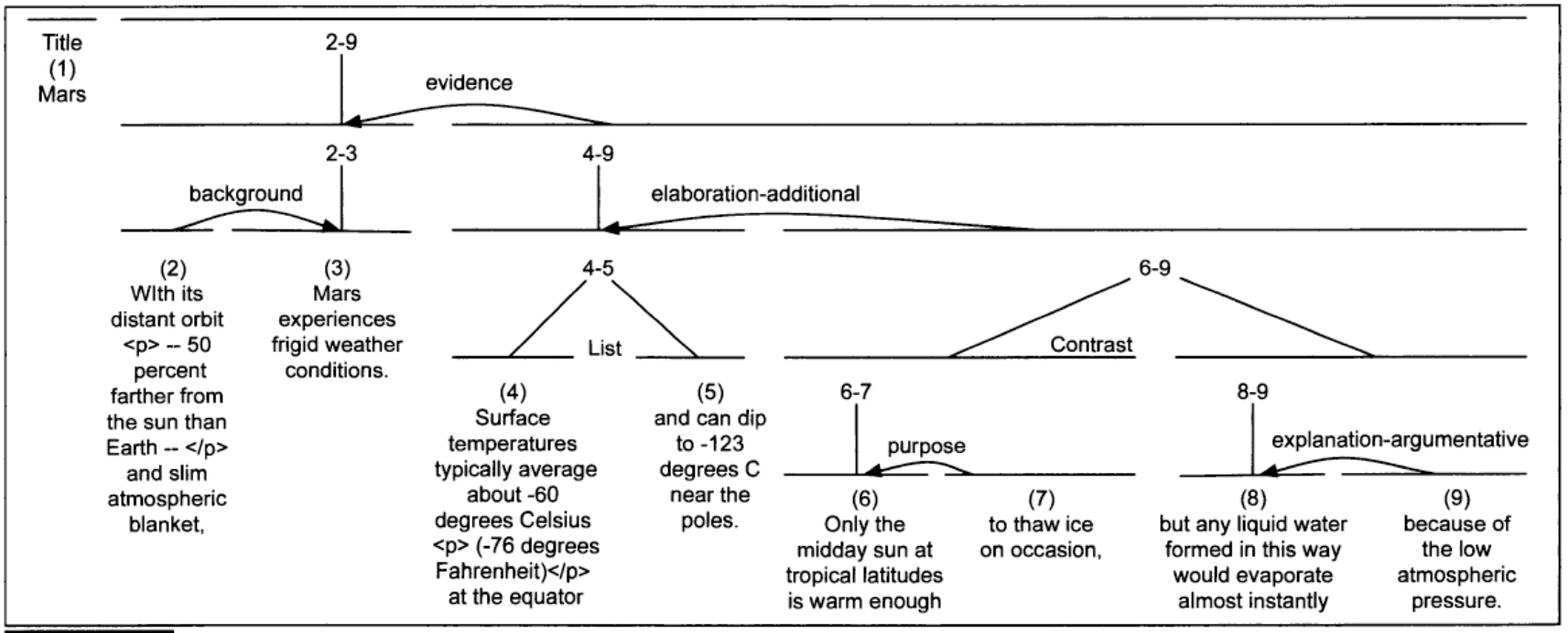


Figure 21.4 A discourse tree for the *Scientific American* text in (21.23), from Marcu (2000a). Note that asymmetric relations are represented with a curved arrow from the satellite to the nucleus.

Jurafsky & Martin

Perceived vs Potential Ambiguity

The Prime Minister announced today that the government intends to trigger Article 50 by the end of this month.

Our analyst George Snowden believes she may do so this week.

- Perceived ambiguity:
  - Is it clear who “she” is?
  - And what “do so” means?
- How many potential antecedents are there for:
  - “she”
  - “do so”?

Extreme Ellipsis: Dialogue

- *British National Corpus KSP 389-393:*

Christine	What have you been up to?
Steve	Nothing.
Michael	Eating.
Leslie	Any phone calls?
Steve	Nah.

- How could we summarise this dialogue?
  - *e.g. C asked what the others had been up to; S said he hadn't been doing anything, M said he'd been eating. L asked whether there had been any phone calls; S said there hadn't been any.*
- ("Summary" is longer than the dialogue …)

Dialogue

- This is what intelligent assistants need to do, e.g. for meeting summarisation:

  A: Well maybe by uh Tuesday you could
  B: Uh-huh
  A: revise the uh
  C: proposal
  B: Mmm Tuesday let's see
  A: and send it around
  B: OK sure sounds good

- How could we summarise this dialogue?
  - *e.g. A suggested (with C) that B could revise the proposal by Tuesday and send it around. B agreed to do that.*
  - *e.g. B agreed to revise the proposal by Tuesday.*

Dialogue Acts

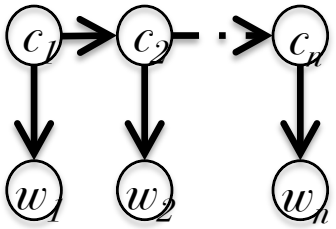
- Speech Acts / Dialogue Acts
  - "How to Do Things with Words" (Searle, 1952)
- Utterances in dialogue are **actions**
  - We ask questions … answer them ...
  - ... greet each other ...
  - ... make promises, threats ...
- And these actions have effects
  - introducing questions for discussion ...
  - ... resolving them ...
  - ... greeting, promising, ...
- We need to keep a record of actions & effects
  - we need them to give a meaningful summary
  - and can (must) use them to build meaning representations
    - (Ginzburg, 1994; 2012)

Dialogue Act Tagging

- Tag utterances with their action type (**dialogue act**):

Christine	What have you been up to?	ASK	WH-Q
Steve	Nothing.	ANSWER	NP-ANS
Michael	Eating.	ANSWER	NP-ANS
Leslie	Any phone calls?	ASK	YN-Q
Steve	Nah.	ANSWER	NEG-ANS

- Sequence modelling task
  - HMMs, CRFs, RNNs
  - Learn from labelled corpus e.g. Switchboard

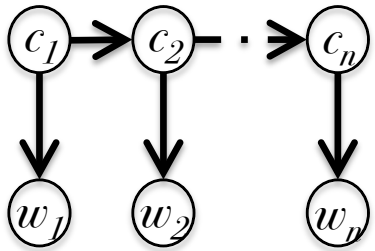


- Need a rich (often domain-dependent) tagset – see e.g. Switchboard corpus:

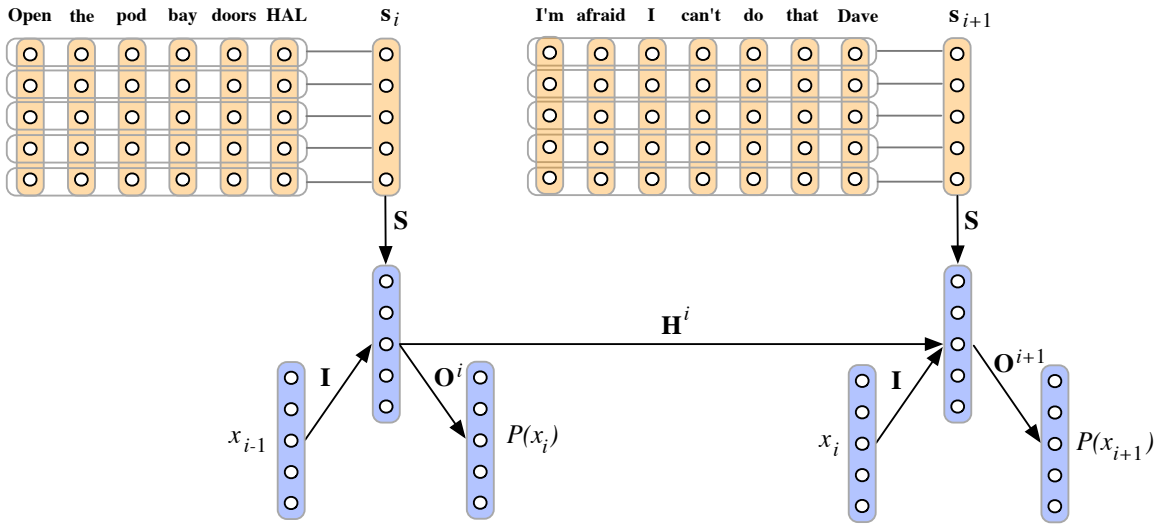
A:	So do you go to college right now?	YN-QUESTION
B:	Yeah	YES-ANSWER
A:	Are yo-	ABANDONED
B:	it's my last year	STATEMENT
A:	What did you say?	CLARIFY
B:	my last year	NP-ANSWER
A:	Oh good for you	APPRECIATION
B:	uh-huh	BACKCHANNEL

Dialogue Act Tagging

- Sequence modelling task
  - HMMs, CRFs as standard approaches
  - Features?
    - Words; syntax; semantics
    - Utterance length, POS patterns
    - Paralinguistic features e.g. intonation?
  - Transition probabilities?
- Needs training from relevant data
  - What corpus?
- Recurrent neural networks
  - Kalchbrenner & Blunsom (2013)



Who's that?	WH-Q
Jim	NP-ANS
Jim?	CLARIFY
Yes	POS-ANS
Is he OK?	YN-Q
No	NEG-ANS



Extreme Ellipsis: Dialogue

- Can resolve ellipsis via "Question Under Discussion (QUD)":
- *British National Corpus KSP 389-393:*

Christine	What have you been up to?	$ask(c,Q)$	$Q = \lambda\{a,x\}.up\_to(a,x)$
Steve	Nothing.	$answer(s,Q(s,n))$	--> $up\_to(s,n)$
Michael	Eating.	$answer(m,Q(m,e))$	--> $up\_to(m,e)$
Leslie	Any phone calls?	$ask(l,Q')$	$Q' = \lambda\{a\}.\exists x.call(x)$
Steve	Nah.	$answer(s,\neg Q'(s))$	--> $\neg\exists x.call(x)$

- But this is still an active research area …
  - (assigning QUD update & attachment structure is hard!)

Extreme Ellipsis: Dialogue

- *British National Corpus KSV 282-285:*

Richard	Oh you're disappointed now aren't you,
	that I, coming back, it's really upset you
Anon 3	That you're coming back?
Richard	Yeah
Anon 3	Yeah

- *British National Corpus KSP 28-32:*

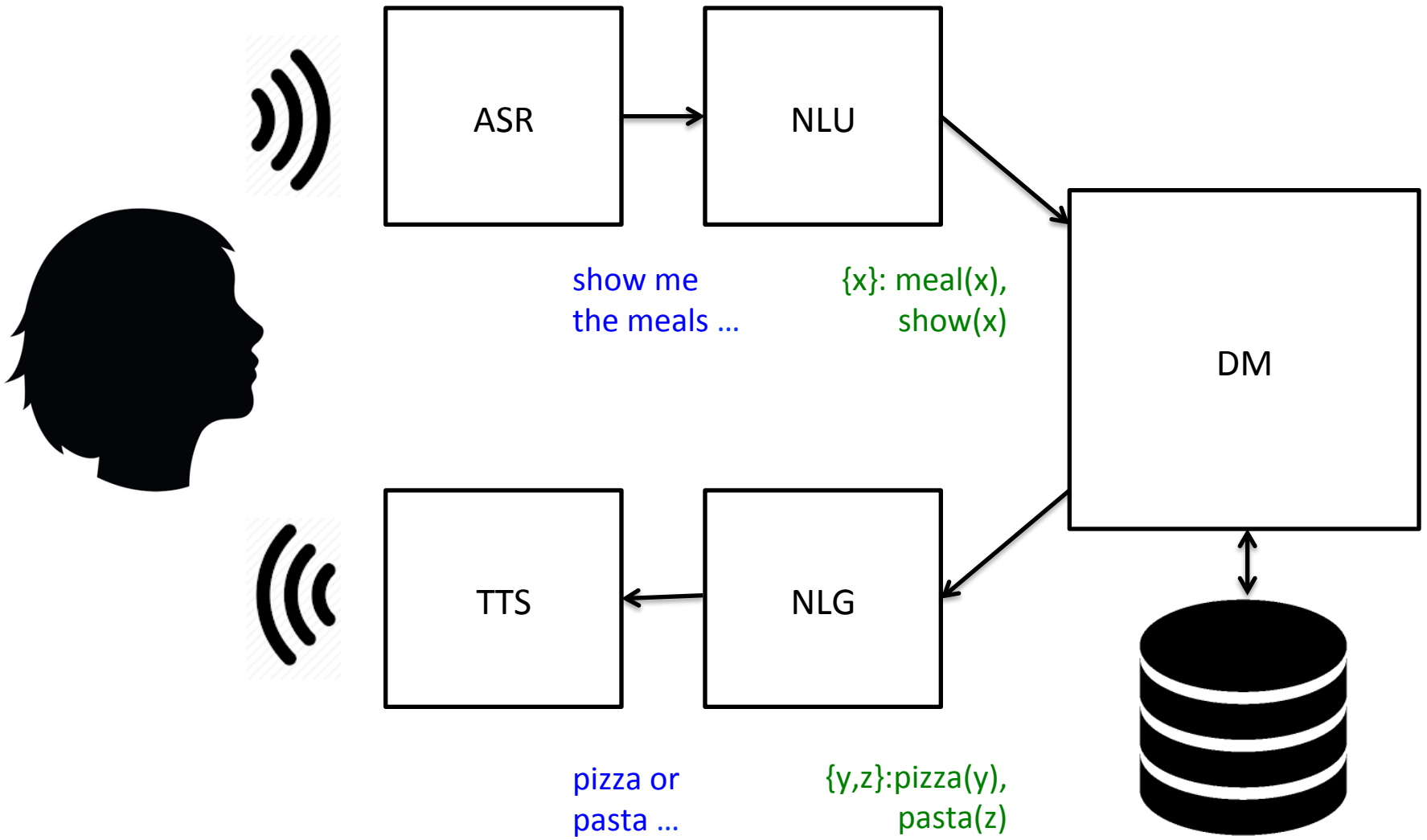
Kevin	Do you er, have you got any whatsername there?
Barry	What?
Kevin	Brochures.
Peter	Brochures?
Unknown	No.
Unknown	No I haven't.

Practical Dialogue Processing

- Human-human dialogue – fairly shallow processing:
  - Dialogue act tagging
  - Topic modelling (& segmentation)
  - Combine DA structures & topics for:
    - Summarisation
    - Decision / action-item detection (e.g. Tur et al, 2010)
    - Dialogue classification/prediction
      - Mental health diagnosis & prediction (e.g. Howes et al 2014)

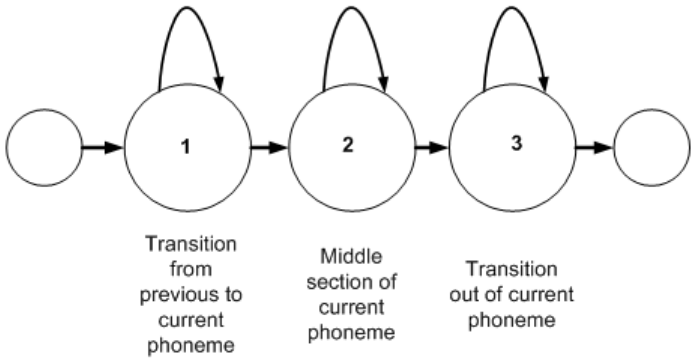
- Human-computer dialogue – can be deeper:
  - More constrained domain & task
    - Higher accuracy, less variation in structure
  - Potential for interaction
    - Clarification, correction, direction & control of structure

Dialogue Systems



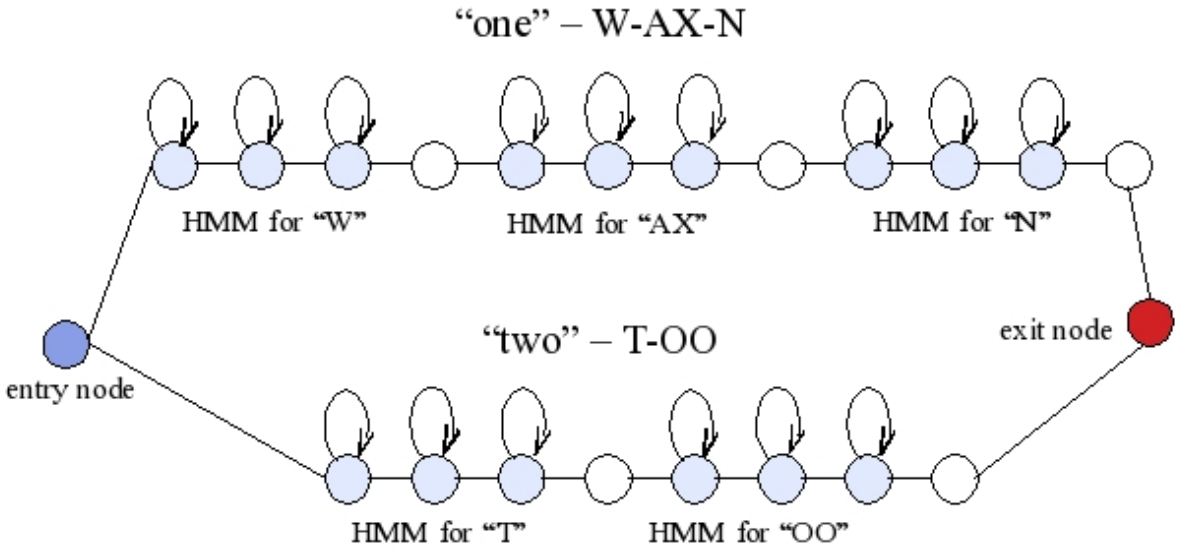
ASR: Automatic Speech Recognition

- Input: acoustic speech signal
- Output: word hypotheses
- Typical error rates high:
  - Google/MS claim 6-8% on benchmarks
  - Often 10-25% in single-user applications
  - Can be c.50% in noisy, multi-party settings
- Sequence modelling task:
  - HMMs / RNNs
  - **Acoustic model:** most likely phoneme sequence
    - e.g. one 3-state (triphone) HMM per phoneme
  - Large alignment problem
    - (where are the state transitions?)



ASR: Automatic Speech Recognition

- Ambiguity: from acoustics only, we can't tell most likely word sequence
  - /ɪtshardtərɜkənaɪzspitʃ/
    - “it's hard to recognise speech”? “it's hard to wreck a nice beach”?
  - /sɪkskwɪd/
    - “six quid”? Or “sick squid”?
  - Are we talking about money or seafood? (or speech or beaches?)
- Second stage: **language model -** most likely word sequence
  - Language modelling tells us about likely words / phrases
  - HMMs combining phoneme models with likely word transitions



ASR Language Modelling

- Language modelling is the part you might have to get involved in
  - Off-the-shelf ASR (e.g. Google Speech API) is good now
  - But often need to train ASR for your language/domain to improve accuracy
- Probabilistic modelling
  - Robust, but needs a lot of transcribed data
- Grammar-based models
  - Much more limited, but you can write them without data
  - Sometimes we **want** a more limited model (constraints)
- Java Speech Grammar Format (Java Speech API)
  - http://java.sun.com/products/java-media/speech/forDevelopers/JSGF/

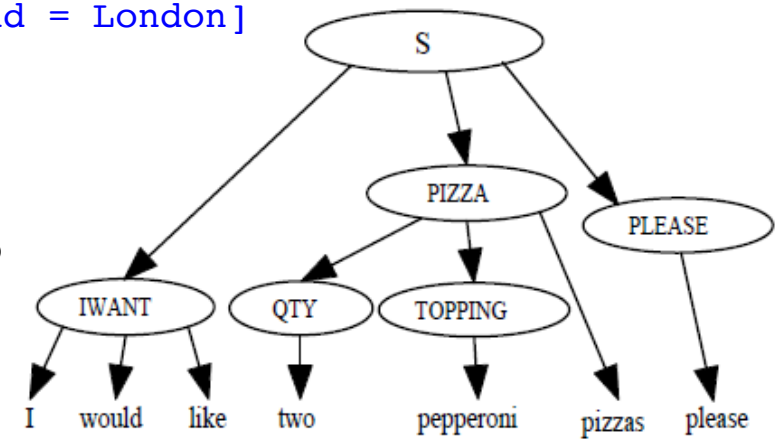
```
public <basicCmd> = <startPolite> <command> <endPolite>;

<startPolite> = (please | kindly | could you) *;
<endPolite> = [ please | thanks | thank you ];

<command> = <action> <object>;
<action> = /10/ open |/2/ close |/1/ delete |/1/ move;
<object> = [the | a] (window | file | menu);
```

NLU: Natural Language Understanding

- This is the part you already know how to do!
  - (and you know how ambiguous/errorful it can be …)
- But you will have many design choices:
  - Representation (LF) format – how deep?

    `[action = go, start = Stockholm, end = London]`

    `[n=2, type=pepperoni]` vs.



  - Parsing method – grammar? HMM? RNN?
  - Knowledge vs data?

- Java Speech API (JSGF) allows:
  - simple keyword-spotting
    - "… delete …" → `[delete]`
  - pattern-matching/slot-filling
    - "I want to (go|fly|…) from {START} to {DEST} [on {DATE}]"
    - → `[start = START, dest = DEST, date = DATE]`

NLG: Natural Language Generation

- The opposite of semantic parsing (NLU):
  - Input = semantic representations
  - Output = word sequences
- In limited domains, usually still template-based
  - "Getting flight details from {START} to {DEST} on {DATE}. One moment please."
  - High-quality, simple
  - But time-consuming to engineer, can be monotonous
- Grammar-based:
  - Use NLU(-like) grammars, generation algorithm
  - More variation
  - Very time-consuming to engineer
- Statistical:
  - Learn e.g. sequence models, RNNs from annotated data
  - More chance of errorful output
  - Need a lot of data

TTS: Speech Synthesis (Text-to-Speech)

- The opposite of ASR:
  - Input = word sequences (with markup?)
  - Output = speech signals
- In principle less difficult than ASR: no search problem
  - (we know what we're trying to say)
- Naturalness and intonation are difficult though
  - Rule-based approaches
    - Mathematical models generate each phoneme
    - e.g. DECTalk (Stephen Hawking)
  - Concatenative approaches
    - Record a sound for each phoneme (actually each **diphone**)
    - Play them back in sequence, with intonation e.g. FreeTTS
  - Fully recorded output
    - Simple, v high quality; but very expensive, inflexible
  - Best systems use a range of units & choose on-the-fly
    - Phones, diphones, words, … to whole sentences
    - e.g. Festival, Cereproc

Dialogue Management

- Communicating with underlying application
  - (ordering train tickets etc)
- Managing communication and error
  - There will be a **lot** of error/ambiguity!
  - Letting the user know what the system can understand
    - Helpful prompts
  - Letting the user know what the system did understand
    - informative & timely responses “searching the flight database ...”
  - Allowing the user to correct errors
    - Telling them when the system didn't understand
- “Grounding”: management of coordination/uncertainty
  - How do humans do this? Backchannels:
    - “uh-huh”, “I see”, “OK”.
    - “Wow!”, “really?”, “no!”
    - “Eh?”, “what do you mean?”, “did you say 'pizza'?”
    - Head nods, eyebrow raising, gaze, gesture (→ screen?)

Backchannels & Clarification

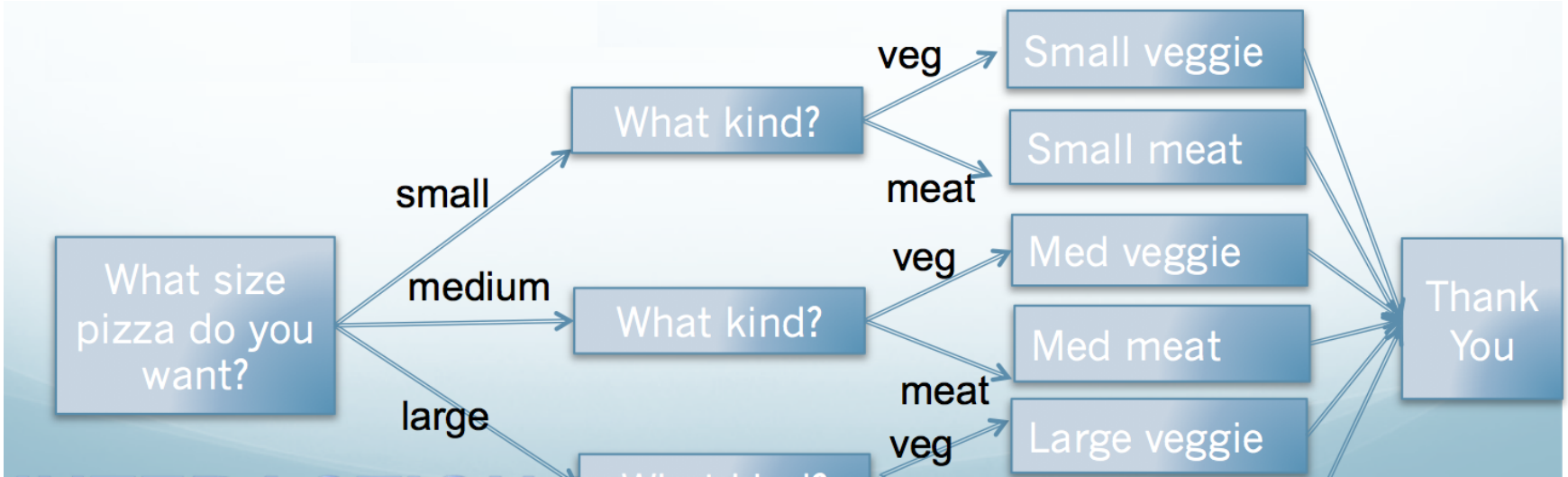
- Show **positive/negative** understanding at critical points
  - After user input
    - ASR & NLU can have very high error rates
  - When there's other processing to do (avoid silence)
    - "OK. Searching the flight database …"
- **Explicitly** indicate problem & level of the problem:
    - "What did you say?"
    - "Did you say "Avatar"?"
    - "I think you said "Avatar", is that correct?"
    - "Which John do you want?"
- **Implicitly** check information
    - "What time do you want to see "Avatar"?"
    - "I found no cinemas showing "Avatar" after 9pm"
    - "The next showing of "Avatar" is at 8pm"
- Common strategy: drive from ASR model confidence
  - Confidence < threshold1: explicit rejections
  - Confidence < threshold2: explicit clarification
  - Otherwise: implicit confirmation in next action

DM Example

- DUDE system [1, 2]
  - Grounding via backchannels
  - Explicit vs implicit confirmation
  - Clarification
- SIRI with errors [3]

Rule-Based DM

- Still the most common in commercial use
  - e.g. VoiceXML
- Dialogue as a graph (i.e. flowchart/script)
  - Path per possible dialogue (including clarification etc)
  - Simple, controllable
  - Supported by standards
  - Only suitable for quite limited interactions



Information-State-Based DM

- Information-state update:
  - Used in many research systems
- Dialogue state as sets of facts, questions, plans etc
  - Still rule-based, but more complex (deep) representations
  - Use of semantic LFs, ellipsis resolution, inference, planning
  - More flexible behaviour
  - More complex to design & maintain
- Probabilistic versions (e.g. Lison, 2015)

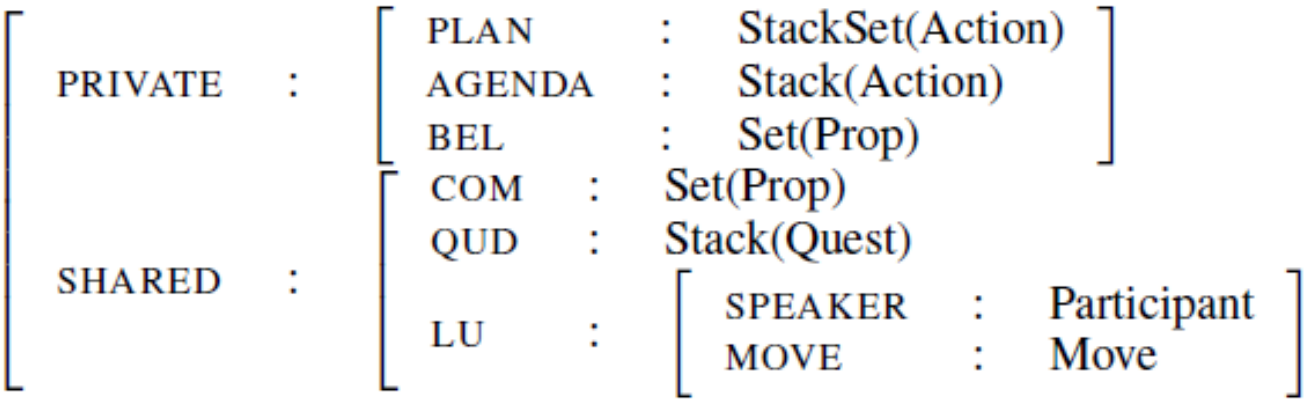
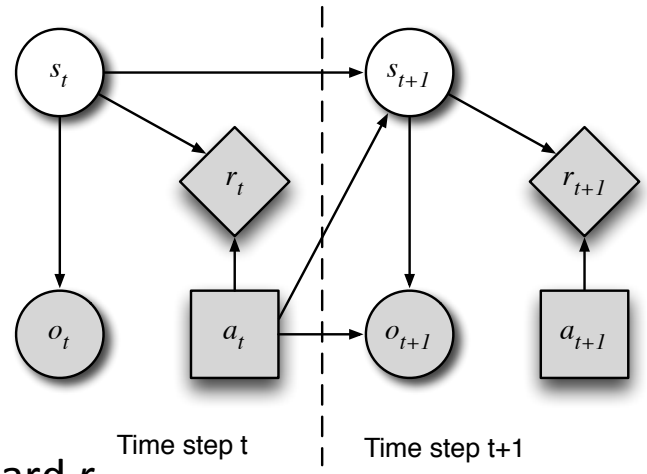


Figure 1: IBiS information state type (Larsson, 2002)

Statistical DM

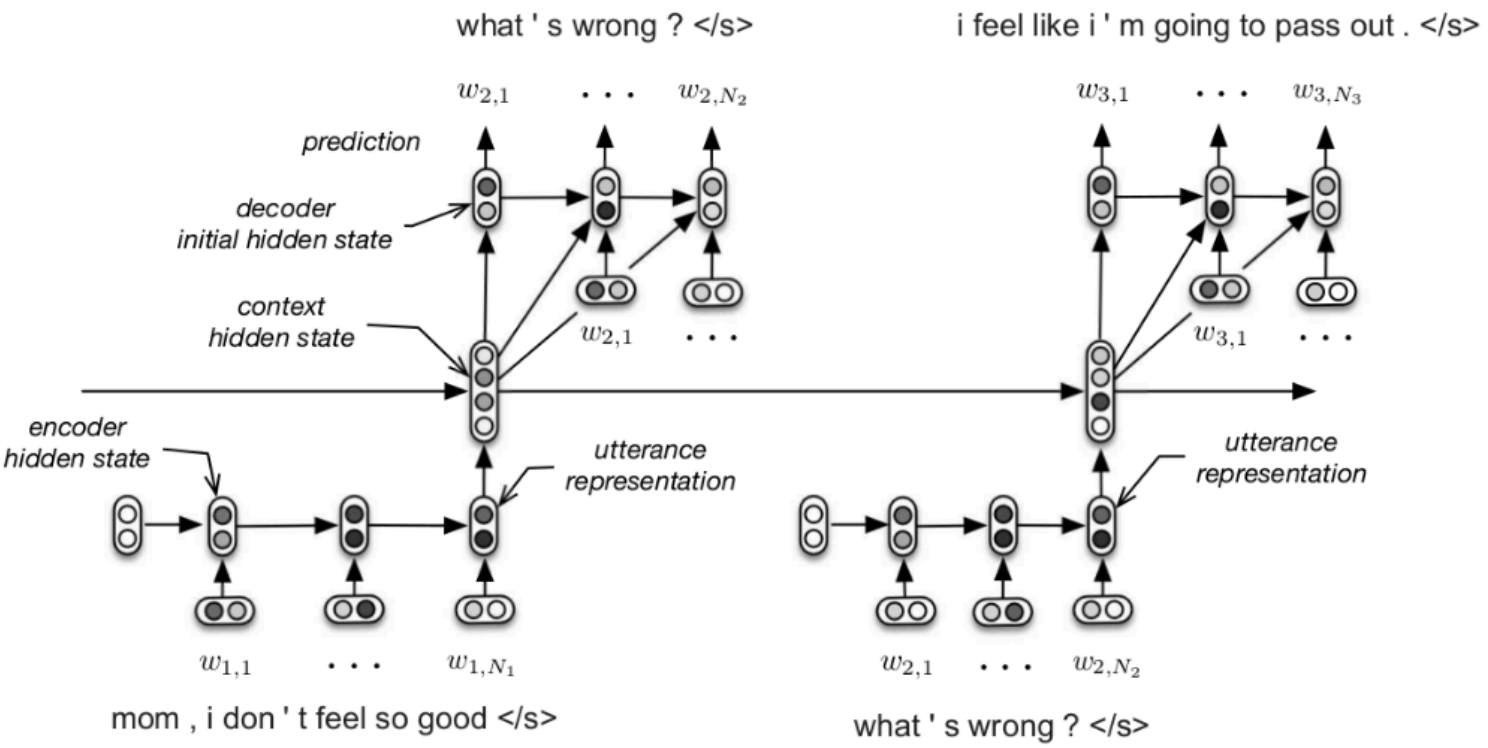
- Probabilistic models
  - Partially Observable Markov Decision Processes (POMDPs)
  - Used in many research systems, some commercial
  - e.g. VocalIQ (Apple)
- Sequence models (extension of HMMs)
  - Observed user moves *o*
  - State represents dialogue "belief" state *s*
    - e.g. destination = Paris, date = 2017-01-03
- Probabilistic decision process
  - Distribution over belief states
  - Emission probabilities $p(o|s)$
  - Transition probabilities $p(s_{t+1}|s_t)$
  - Take optimal system action *a* given expected reward *r*
- Trained from data **interactively**
  - Reinforcement learning
  - Explore possible system decision paths
  - Learn which led to good outcomes



Young et al (2013)

End-to-End Systems

- Increasing work in “end-to-end” (non-modular) systems
  - e.g. hierarchical recurrent NNs (Serban et al, 2015)
- Entirely data-driven
  - Robust; but data-hungry and non-modular



Training

- Where do we get data from?
- Annotated existing dialogues
  - e.g. Switchboard corpus
  - Good for general dialogue act tagging
  - But limited:
    - We often need domain/system-specific data
    - No use for POMDP training
    - (Dialogues can go in many different directions)
- Wizard-of-Oz studies
  - Gather data using humans as simulated systems
  - Good for small datasets, and for system prototyping/evaluation
- Reinforcement learning needs thousands/millions of interactions
  - User simulations
  - Train simulated user (e.g. DA n-gram model)
  - Use in probabilistic training

Evaluation

- Task-level evaluation metrics
  - Efficiency: elapsed time, system turns, user turns
  - Quality: mean recognition/understanding scores, timeouts, rejections, helps, cancels etc.
  - Task success: database query completion rate etc.
- User satisfaction metrics
  - Survey-based e.g. 5-point Likert scale questionnaire
  - Harder to get, harder to pinpoint individual components
  - But this is what we really want to know …
- PARADISE method (Walker et al, 1998)
  - Measure:
    - (a) module/task-level metrics
    - (b) User surveys on same data
  - Train linear regression model to predict (b) from (a)