Dialog Management & Dialog Acts

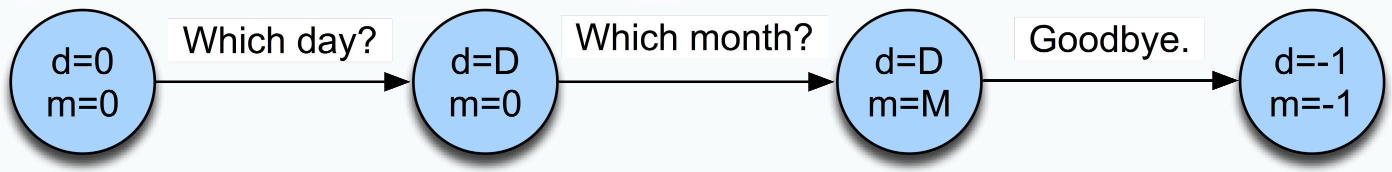
Ling575
Spoken Dialog
May 1, 2013

2 possible policies

Strategy 1 is better than strategy 2 when improved error rate justifies longer interaction:

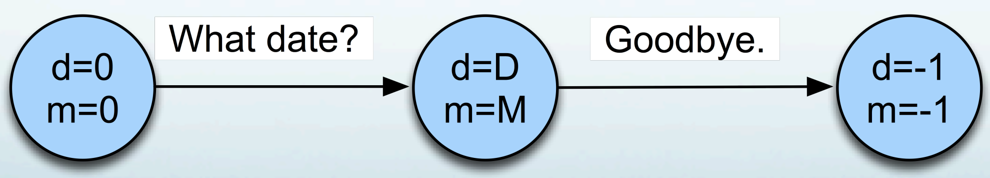
$$p_o - p_d > \frac{w_i}{2w_e}$$

Policy 1 (directive)



$c_1 = -3w_i + 2p_d w_e$

Policy 2 (open)



$c_2 = -2w_i + 2p_o w_e$

That was an easy optimization

- Only two actions, only tiny # of policies
- In general, number of actions, states, policies is quite large
- So finding optimal policy π^* is harder
- We need reinforcement learning
- Back to MDPs:

MDP

- We can think of a dialogue as a trajectory in state space

$$s_1 \rightarrow_{a1,r1} s_2 \rightarrow_{a2,r2} s_3 \rightarrow_{a3,r3} \cdots$$

- The best policy π* is the one with the greatest expected reward over all trajectories
- How to compute a reward for a state sequence?

Reward for a state sequence

- One common approach: discounted rewards
- Cumulative reward Q of a sequence is discounted sum of utilities of individual states

$$Q([s_0, a_0, s_1, a_1, s_2, a_2 \cdots]) = R(s_0, a_0) + \gamma R(s_1, a_1) + \gamma^2 R(s_2, a_2) + \cdots,$$

- Makes agent care more about current than future rewards; the more future a reward, the more discounted its value

The Markov assumption

- MDP assumes that state transitions are Markovian

$$P(s_{t+1} \mid s_t, s_{t-1}, \ldots, s_o, a_t, a_{t-1}, \ldots, a_o) = P_T(s_{t+1} \mid s_t, a_t)$$

Expected reward for an action

- Expected cumulative reward Q(s,a) for taking a particular action from a particular state can be computed by Bellman equation:

$$Q(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) \max_{a'} Q(s',a')$$

- Expected cumulative reward for a given state/action pair is:
  - immediate reward for current state
  - + expected discounted utility of all possible next states s'
  - Weighted by probability of moving to that state s'
  - And assuming once there we take optimal action a'

What we need for Bellman equation

- A model of p(s' |s,a)
- Estimate of R(s,a)
- How to get these?

What we need for Bellman equation

- A model of $p(s'|s,a)$
- Estimate of $R(s,a)$
- How to get these?
- If we had labeled training data
  - $P(s'|s,a) = C(s,s',a)/C(s,a)$

What we need for Bellman equation

- A model of p(s' |s,a)
- Estimate of R(s,a)
- How to get these?
- If we had labeled training data
  - P(s' |s,a) = C(s,s' ,a)/C(s,a)
- If we knew the final reward for whole dialogue R(s1,a1,s2,a2,…,sn)
- Given these parameters, can use value iteration algorithm to learn Q values (pushing back reward values over state sequences) and hence best policy

Final reward

- What is the final reward for whole dialogue R(s1,a1,s2,a2,…,sn)?
- This is what our automatic evaluation metric PARADISE computes!
- The general goodness of a whole dialogue!!!!!

How to estimate $p(s'|s,a)$ without labeled data

How to estimate p(s' |s,a) without labeled data

- Have random conversations with real people
  - Carefully hand-tune small number of states and policies
  - Then can build a dialogue system which explores state space by generating a few hundred random conversations with real humans
  - Set probabilities from this corpus

How to estimate p(s' |s,a) without labeled data

- Have random conversations with real people
  - Carefully hand-tune small number of states and policies
  - Then can build a dialogue system which explores state space by generating a few hundred random conversations with real humans
  - Set probabilities from this corpus
- Have random conversations with simulated people
  - Now you can have millions of conversations with simulated people
  - So you can have a slightly larger state space

An example

- Singh, S., D. Litman, M. Kearns, and M. Walker. 2002. Optimizing Dialogue Management with Reinforcement Learning: Experiments with the NJFun System. Journal of AI Research.

- NJFun system, people asked questions about recreational activities in New Jersey
- Idea of paper: use reinforcement learning to make a small set of optimal policy decisions

Very small # of states and acts

- **States**: specified by values of 8 features
  - Which slot in frame is being worked on (1-4)
  - ASR confidence value (0-5)
  - How many times a current slot question had been asked
  - Restrictive vs. non-restrictive grammar
  - Result: 62 states
- **Actions**: each state only 2 possible actions
  - Asking questions: System versus user initiative
  - Receiving answers: explicit versus no confirmation.

Ran system with real users

- 311 conversations
- Simple binary reward function
  - 1 if competed task (finding museums, theater, winetasting in NJ area)
  - 0 if not
- System learned good dialogue strategy: Roughly
  - Start with user initiative
  - Backoff to mixed or system initiative when re-asking for an attribute
  - Confirm only a lower confidence values

State of the art

- Only a few such systems
  - From (former) ATT Laboratories researchers, now dispersed
  - And Cambridge UK lab
- Hot topics:
  - Partially observable MDPs (POMDPs)
  - We don't REALLY know the user's state (we only know what we THOUGHT the user said)
  - So need to take actions based on our BELIEF , I.e. a probability distribution over states rather than the "true state"

Summary

- Utility-based conversational agents
  - Policy/strategy for:
    - Confirmation
    - Rejection
    - Open/directive prompts
    - Initiative
    - +?????
  - MDP
  - POMDP

Roadmap

- Dialog acts
  - Annotation
    - Basic dialog acts & tagsets
    - Reliability
  - Recognition
    - Approaches & information
    - N-gram DA tagging
    - Feature Latent Semantic Analysis
    - SVMs with HMMs

Dialogue Acts

- Extension of speech acts
  - Adds structure related to conversational phenomena
    - Grounding, adjacency pairs, etc

Dialogue Acts

- Extension of speech acts
  - Adds structure related to conversational phenomena
    - Grounding, adjacency pairs, etc
- Many proposed tagsets
  - Verbmobil: acts specific to meeting sched domain

Dialogue Acts

- Extension of speech acts
  - Adds structure related to conversational phenomena
    - Grounding, adjacency pairs, etc
- Many proposed tagsets
  - Verbmobil: acts specific to meeting sched domain
  - DAMSL: Dialogue Act Markup in Several Layers
    - Forward looking functions: speech acts
    - Backward looking function: grounding, answering

Dialogue Acts

- Extension of speech acts
  - Adds structure related to conversational phenomena
    - Grounding, adjacency pairs, etc
- Many proposed tagsets
  - Verbmobil: acts specific to meeting sched domain
  - DAMSL: Dialogue Act Markup in Several Layers
    - Forward looking functions: speech acts
    - Backward looking function: grounding, answering
  - Conversation acts:
    - Add turn-taking and argumentation relations

Verbmobil DA

- 18 high level tags

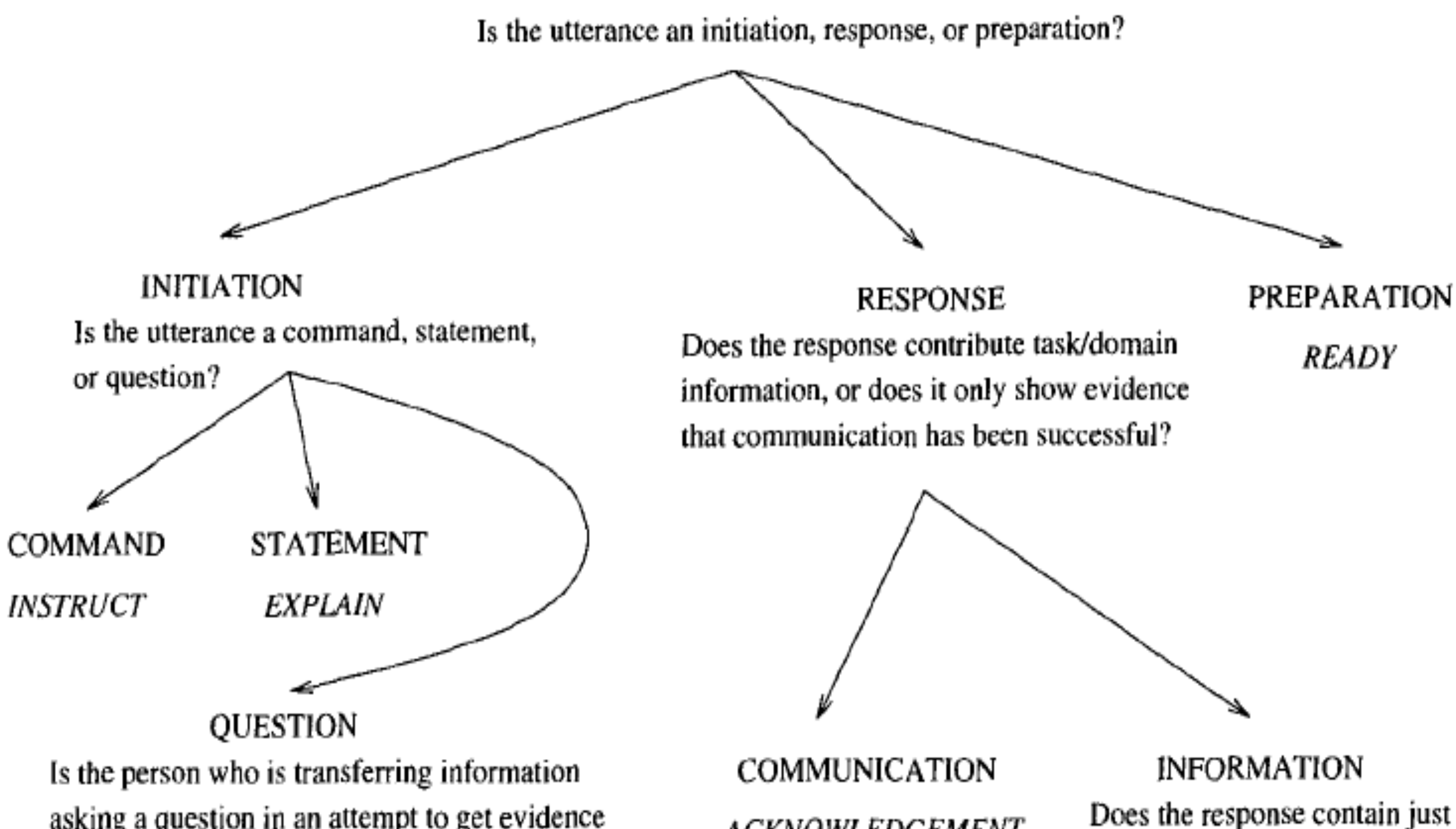
Tag	Example
THANK	*Thanks*
GREET	*Hello Dan*
INTRODUCE	*It's me again*
BYE	*Allright bye*
REQUEST-COMMENT	*How does that look?*
SUGGEST	*from thirteenth through seventeenth June*
REJECT	*No Friday I'm booked all day*
ACCEPT	*Saturday sounds fine,*
REQUEST-SUGGEST	*What is a good day of the week for you?*
INIT	*I wanted to make an appointment with you*
GIVE_REASON	*Because I have meetings all afternoon*
FEEDBACK	*Okay*
DELIBERATE	*Let me check my calendar here*
CONFIRM	*Okay, that would be wonderful*
CLARIFY	*Okay, do you mean Tuesday the 23rd?*
DIGRESS	*[we could meet for lunch] and eat lots of ice cream*
MOTIVATE	*We should go to visit our subsidiary in Munich*
GARBAGE	*Oops, I-*

Figure 24.17 The 18 high-level dialogue acts used in Verbmobil-1, abstracted over a total of 43 more specific dialogue acts. Examples are from Jekat et al. (1995).

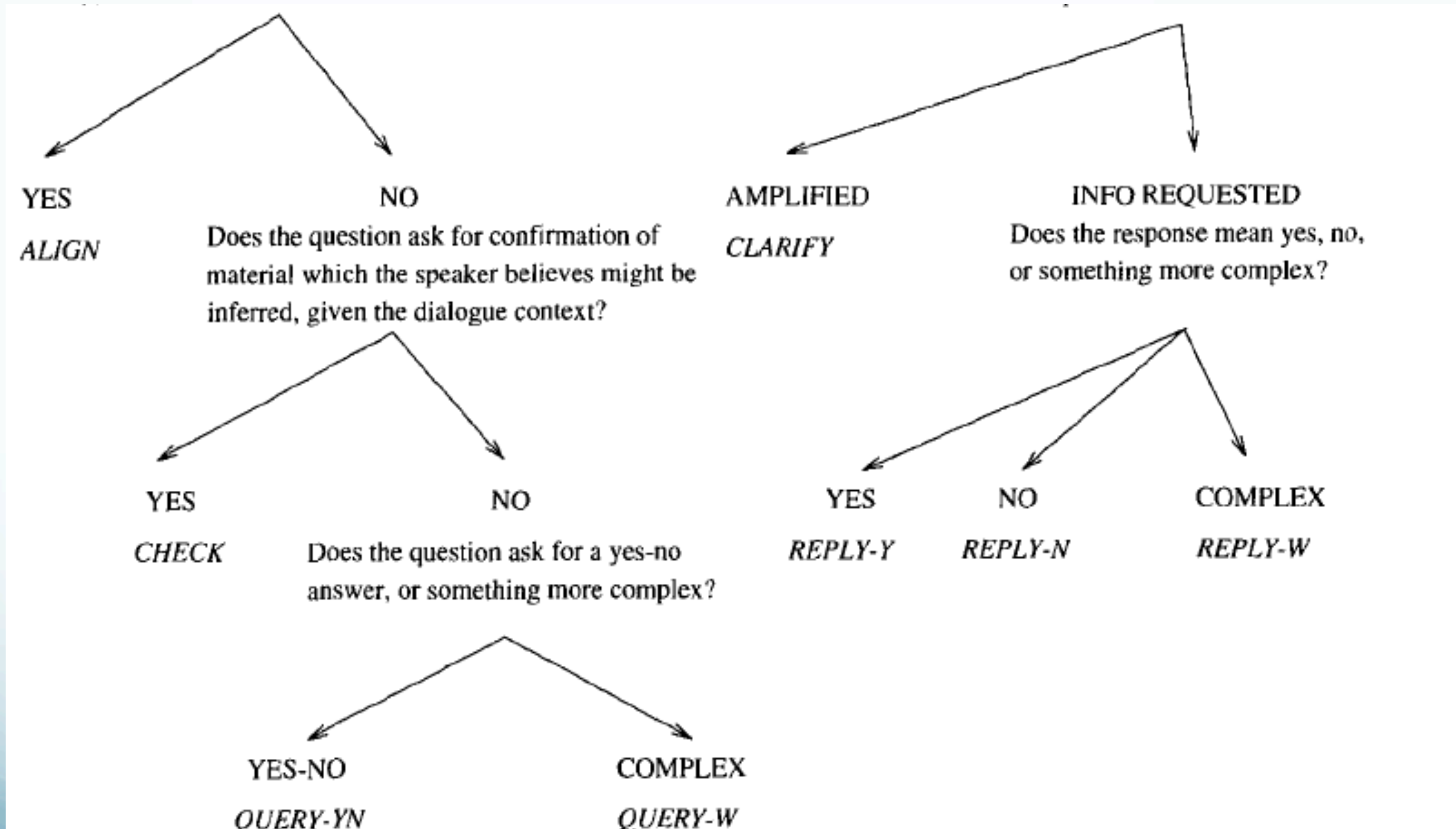
Maptask: Dialog act tagging & analysis

- Goal:
  - Dialog structure coding that is:
    - Task-independent: applicable to human or machine
    - Linked to higher-levels of discourse structure
    - Generic: Interoperate with other models
- Overall model: 3 levels
  - Transactions: Subdialog accomplishing major task step
  - Games: Discourse segments of initiations/responses
  - Moves: Individual initiations or responses
    - Adjacency pairs

Dialog Acts


Is the utterance an initiation, response, or preparation?
INITIATION
Is the utterance a command, statement, or question?
RESPONSE
Does the response contribute task/domain information, or does it only show evidence that communication has been successful?
PREPARATION
READY
COMMAND
INSTRUCT
STATEMENT
EXPLAIN
QUESTION
Is the person who is transferring information asking a question in an attempt to get evidence
COMMUNICATION
INFORMATION
Does the response contain just

Dialog Acts


YES
ALIGN
NO
Does the question ask for confirmation of material which the speaker believes might be inferred, given the dialogue context?
AMPLIFIED
CLARIFY
INFO REQUESTED
Does the response mean yes, no, or something more complex?
YES
CHECK
NO
Does the question ask for a yes-no answer, or something more complex?
YES
REPLY-Y
NO
REPLY-N
COMPLEX
REPLY-W
YES-NO
QUERY-YN
COMPLEX
QUERY-W

Maptask Scenario

- Two participants:
  - Giver and follower
- Each has a map, differing in detail
  - Giver has a route
- Goal: Follower replicates route on own map
  - Requires clarifications, naming, etc

Dialog Act Inventory

- Instruct: command other to do something
- Explain: state information not explicitly requested
- Check: ask for confirmation
- Align: check other's attn, agreement, readiness: Ok?
- Query YN; Query-W: yes/no, other question
- Acknowledge: indicate heard and understood
- Reply-Y; Reply-N; Reply-W:
- Clarify: reply beyond what was asked
- Ready: after completion of one game, before start of other

Interrater Agreement

- How good is tagging? A tagset?

Interrater Agreement

- How good is tagging? A tagset?
- Criterion: How accurate/consistent is it?

Interrater Agreement

- How good is tagging? A tagset?
- Criterion: How accurate/consistent is it?
- Stability:
  - Is the same rater self-consistent?

Interrater Agreement

- How good is tagging? A tagset?
- Criterion: How accurate/consistent is it?
- Stability:
  - Is the same rater self-consistent?
- Reproducibility:
  - Do multiple annotators agree with each other?

Interrater Agreement

- How good is tagging? A tagset?
- Criterion: How accurate/consistent is it?
- Stability:
  - Is the same rater self-consistent?
- Reproducibility:
  - Do multiple annotators agree with each other?
- Accuracy:
  - How well do coders agree with some "gold standard"?

Agreement Measure

- Kappa (K) coefficient

Agreement Measure

- Kappa (K) coefficient
  - Applies to classification into discrete categories

Agreement Measure

- Kappa (K) coefficient
  - Applies to classification into discrete categories
  - Corrects for chance agreement
    - K<0 : agree less than expected by chance

Agreement Measure

- Kappa (K) coefficient
  - Applies to classification into discrete categories
  - Corrects for chance agreement
    - $K<0$: agree less than expected by chance
  - Quality intervals:
    - $>= 0.8$: Very good; $0.6<K<0.8$: Good, etc

Agreement Measure

- Kappa (K) coefficient
  - Applies to classification into discrete categories
  - Corrects for chance agreement
    - K<0 : agree less than expected by chance
  - Quality intervals:
    - >= 0.8: Very good; 0.6<K<0.8: Good, etc
- Maptask: K=0.92 on segmentation,
  - K = 0.83 on move labels – 13 tags

Dialogue Act Interpretation

- Automatically tag utterances in dialogue

Dialogue Act Ambiguity

- Indirect speech acts

A		I was wanting to make some arrangements for a trip that I'm going to be taking uh to LA uh beginning of the week after next.
B		OK uh let me pull up your profile and I'll be right with you here. [pause]
B		**And you said you wanted to travel next week?**
A		Uh yes.

Dialogue Act Ambiguity

- Indirect speech acts

A	OPEN-OPTION	I was wanting to make some arrangements for a trip that I'm going to be taking uh to LA uh beginning of the week after next.
B		OK uh let me pull up your profile and I'll be right with you here. [pause]
B		**And you said you wanted to travel next week?**
A		Uh yes.

Dialogue Act Ambiguity

- Indirect speech acts

A	OPEN-OPTION	I was wanting to make some arrangements for a trip that I'm going to be taking uh to LA uh beginning of the week after next.
B	HOLD	OK uh let me pull up your profile and I'll be right with you here. [pause]
B		**And you said you wanted to travel next week?**
A		Uh yes.

Dialogue Act Ambiguity

- Indirect speech acts

A	OPEN-OPTION	I was wanting to make some arrangements for a trip that I'm going to be taking uh to LA uh beginning of the week after next.
B	HOLD	OK uh let me pull up your profile and I'll be right with you here. [pause]
B	CHECK	**And you said you wanted to travel next week?**
A		Uh yes.

Dialogue Act Ambiguity

- Indirect speech acts

A	OPEN-OPTION	I was wanting to make some arrangements for a trip that I'm going to be taking uh to LA uh beginning of the week after next.
B	HOLD	OK uh let me pull up your profile and I'll be right with you here. [pause]
B	CHECK	**And you said you wanted to travel next week?**
A	ACCEPT	Uh yes.

Plan-inference-based

- Classic AI (BDI) planning framework
  - Model Belief, Knowledge, Desire
    - Formal definition with predicate calculus
      - Axiomatization of plans and actions as well
      - STRIPS-style: Preconditions, Effects, Body
  - Rules for plan inference

Plan-inference-based

- Classic AI (BDI) planning framework
  - Model Belief, Knowledge, Desire
    - Formal definition with predicate calculus
      - Axiomatization of plans and actions as well
      - STRIPS-style: Preconditions, Effects, Body
  - Rules for plan inference
- Elegant, but..
  - Labor-intensive rule, KB, heuristic development
  - Effectively AI-complete

Cue-based Interpretation

- Employs sets of features to identify
  - Words and collocations: Please -> request
  - Prosody: Rising pitch -> yes/no question
  - Conversational structure: prior act
- Example: Check:
  - Syntax: tag question ",right?"
  - Syntax + prosody: Fragment with rise
  - N-gram: argmax d P(d)P(W|d)
    - So you, sounds like, etc

Dialogue Act Recognition

- How can we classify dialogue acts?
- Sources of information:

Dialogue Act Recognition

- How can we classify dialogue acts?
- Sources of information:
  - Word information:
    - *Please, would you*: request; *are you*: yes-no question

Dialogue Act Recognition

- How can we classify dialogue acts?
- Sources of information:
  - Word information:
    - *Please, would you*: request; *are you*: yes-no question
    - N-gram grammars
  - Prosody:

Dialogue Act Recognition

- How can we classify dialogue acts?
- Sources of information:
  - Word information:
    - *Please, would you*: request; *are you*: yes-no question
    - N-gram grammars
  - Prosody:
    - Final rising pitch: question; final lowering: statement
    - Reduced intensity: *Yeah:* agreement vs backchannel

Dialogue Act Recognition

- How can we classify dialogue acts?
- Sources of information:
  - Word information:
    - *Please, would you*: request; *are you*: yes-no question
    - N-gram grammars
  - Prosody:
    - Final rising pitch: question; final lowering: statement
    - Reduced intensity: *Yeah:* agreement vs backchannel
  - Adjacency pairs:

Dialogue Act Recognition

- How can we classify dialogue acts?
- Sources of information:
  - Word information:
    - *Please, would you*: request; *are you*: yes-no question
    - N-gram grammars
  - Prosody:
    - Final rising pitch: question; final lowering: statement
    - Reduced intensity: *Yeah:* agreement vs backchannel
  - Adjacency pairs:
    - Y/N question, agreement vs Y/N question, backchannel
    - DA bi-grams

Task & Corpus

- Goal:
  - Identify dialogue acts in conversational speech

Task & Corpus

- Goal:
  - Identify dialogue acts in conversational speech
- Spoken corpus: Switchboard
  - Telephone conversations between strangers
  - Not task oriented; topics suggested
  - 1000s of conversations
    - recorded, transcribed, segmented

Dialogue Act Tagset

- Cover general conversational dialogue acts
  - No particular task/domain constraints

Dialogue Act Tagset

- Cover general conversational dialogue acts
  - No particular task/domain constraints
- Original set: ~50 tags
  - Augmented with flags for task, conv mgmt
    - 220 tags in labeling: some rare

Dialogue Act Tagset

- Cover general conversational dialogue acts
  - No particular task/domain constraints
- Original set: ~50 tags
  - Augmented with flags for task, conv mgmt
    - 220 tags in labeling: some rare
- Final set: 42 tags, mutually exclusive
  - SWBD-DAMSL
  - Agreement: K=0.80 (high)

Dialogue Act Tagset

- Cover general conversational dialogue acts
  - No particular task/domain constraints
- Original set: ~50 tags
  - Augmented with flags for task, conv mgmt
    - 220 tags in labeling: some rare
- Final set: 42 tags, mutually exclusive
  - SWBD-DAMSL
  - Agreement: K=0.80 (high)
- 1,155 conv labeled: split into train/test

Common Tags

- **Statement** & **Opinion**: declarative +/- op
- **Question**: Yes/No&Declarative: form, force
- **Backchannel:** Continuers like uh-huh, yeah
- **Turn Exit/Adandon:** break off, +/- pass
- **Answer :** Yes/No, follow questions
- **Agreement**: Accept/Reject/Maybe

Probabilistic Dialogue Models

- HMM dialogue models

Probabilistic Dialogue Models

- HMM dialogue models
  - States = Dialogue acts; Observations: Utterances
    - Assume decomposable by utterance
    - Evidence from true words, ASR words, prosody

$$d^* = \operatorname*{argmax}_d P(d \mid o) = \operatorname*{argmax}_d \frac{P(o \mid d)P(d)}{P(o)} = \operatorname*{argmax}_d P(o \mid d)P(d)$$

Probabilistic Dialogue Models

- HMM dialogue models
  - States = Dialogue acts; Observations: Utterances
    - Assume decomposable by utterance
    - Evidence from true words, ASR words, prosody

$$d^* = \underset{d}{\operatorname{argmax}} P(d \mid o) = \underset{d}{\operatorname{argmax}} \frac{P(o \mid d) P(d)}{P(o)} = \underset{d}{\operatorname{argmax}} P(o \mid d) P(d)$$

$$P(o \mid d) = P(f \mid d) P(W \mid d)$$

Probabilistic Dialogue Models

- HMM dialogue models
  - States = Dialogue acts; Observations: Utterances
    - Assume decomposable by utterance
    - Evidence from true words, ASR words, prosody

$$d^* = \operatorname*{argmax}_d P(d \mid o) = \operatorname*{argmax}_d \frac{P(o \mid d)P(d)}{P(o)} = \operatorname*{argmax}_d P(o \mid d)P(d)$$

$$P(o \mid d) = P(f \mid d)P(W \mid d)$$

$$P(W \mid d) = \prod_{i=2}^{N} P(w_i \mid w_{i-1}, w_{i-2} \ldots w_{i-N+1}, d)$$

Probabilistic Dialogue Models

- HMM dialogue models
  - States = Dialogue acts; Observations: Utterances
    - Assume decomposable by utterance
    - Evidence from true words, ASR words, prosody

$$d^* = \underset{d}{\operatorname{argmax}} P(d \mid o) = \underset{d}{\operatorname{argmax}} \frac{P(o \mid d)P(d)}{P(o)} = \underset{d}{\operatorname{argmax}} P(o \mid d)P(d)$$

$$P(o \mid d) = P(f \mid d)P(W \mid d)$$

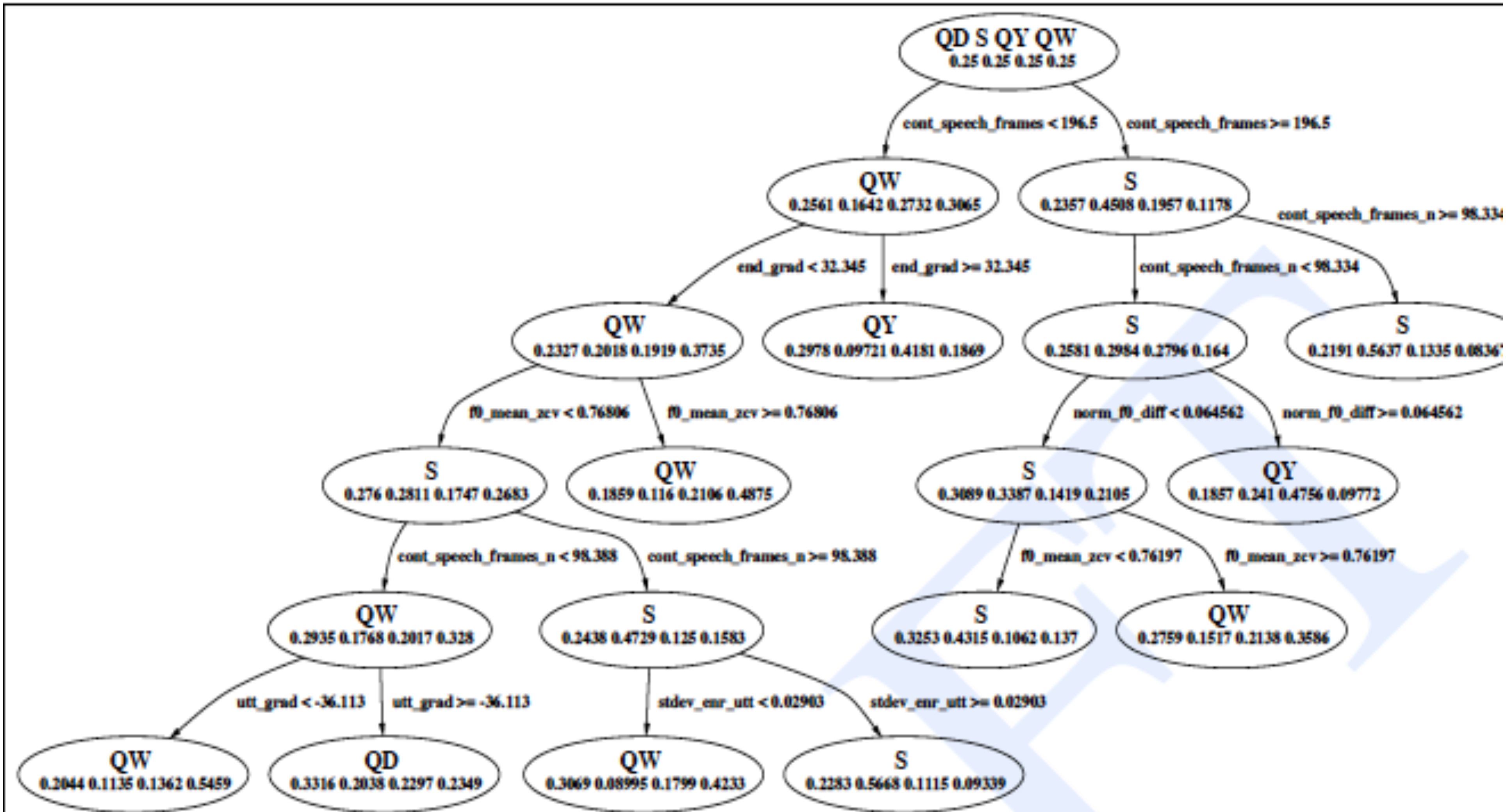
$$P(W \mid d) = \prod_{i=2}^{N} P(w_i \mid w_{i-1}, w_{i-2} \ldots w_{i-N+1}, d)$$

$$d^* = \underset{d}{\operatorname{argmax}} P(d \mid d_{t-1})P(f \mid d)P(W \mid d)$$

DA Classification - Prosody

- Features:
  - Duration, pause, pitch, energy, rate, gender
    - Pitch accent, tone
- Results:
  - Decision trees: 5 common classes
    - 45.4% - baseline=16.6%

Prosodic Decision Tree



DA Classification -Words

- Words
  - Combines notion of discourse markers and collocations:
    - e.g. uh-huh=Backchannel
  - Contrast: true words, ASR 1-best, ASR n-best
- Results:
  - Best: 71%- true words, 65% ASR 1-best

DA Classification - All

- Combine word and prosodic information
  - Consider case with ASR words and acoustics

DA Classification - All

- Combine word and prosodic information
  - Consider case with ASR words and acoustics
  - Prosody classified by decision trees
    - Incorporate decision tree posteriors in model for $P(f|d)$

DA Classification - All

- Combine word and prosodic information
  - Consider case with ASR words and acoustics
  - Prosody classified by decision trees
    - Incorporate decision tree posteriors in model for P(f|d)

$$d^* = P(d \mid d_{t-1}) \frac{P(d \mid f)}{P(d)} \prod_{i=2}^{N} P(w_i \mid w_{i-1} \ldots w_{i-N+1}, d)$$

- Slightly better than raw ASR

Integrated Classification

- Focused analysis
  - Prosodically disambiguated classes
    - Statement/Question-Y/N and Agreement/Backchannel
    - Prosodic decision trees for agreement vs backchannel
      - Disambiguated by duration and loudness

Integrated Classification

- Focused analysis
  - Prosodically disambiguated classes
    - Statement/Question-Y/N and Agreement/Backchannel
    - Prosodic decision trees for agreement vs backchannel
      - Disambiguated by duration and loudness
  - Substantial improvement for prosody+words
    - True words: S/Q: 85.9%-> 87.6; A/B: 81.0%->84.7

Integrated Classification

- Focused analysis
  - Prosodically disambiguated classes
    - Statement/Question-Y/N and Agreement/Backchannel
    - Prosodic decision trees for agreement vs backchannel
      - Disambiguated by duration and loudness
  - Substantial improvement for prosody+words
    - True words: S/Q: 85.9%-> 87.6; A/B: 81.0%->84.7
    - ASR words: S/Q: 75.4%->79.8; A/B: 78.2%->81.7
  - More useful when recognition is iffy

Dialog Act Tagging with Feature Latent Semantic Analysis

Latent Semantic Analysis (LSA)

- Dumais, Deerwester (1990)
- Latent semantic classes (topics)

FLSA slides courtesy Irena Matveeva

Latent Semantic Analysis (LSA)

- Dumais, Deerwester (1990)
- Latent semantic classes (topics)
- Input: term-document matrix D

documents are vectors in the vocabulary space

FLSA slides courtesy Irena Matveeva

Latent Semantic Analysis (LSA)

- Dumais, Deerwester (1990)
- Latent semantic classes (topics)
- Input: term-document matrix D

documents are vectors in the vocabulary space

- Output: modified matrix D'

documents are vectors in the latent semantic space

FLSA slides courtesy Irena Matveeva

Latent Semantic Analysis (LSA)

- Dumais, Deerwester (1990)
- Latent semantic classes (topics)
- Input: term-document matrix D

documents are vectors in the vocabulary space

- Output: modified matrix D'

documents are vectors in the latent semantic space

- Use D' for classification

FLSA slides courtesy Irena Matveeva

Latent Semantic Analysis (LSA)

- $D = USV^T$

$d = (w_1, \ldots, w_N)$

Latent Semantic Analysis (LSA)

- $D=USV^T$

$d=(w_1, \ldots, w_N)$

- $D'=US_kV^T$

$d=(z_1,\ldots,z_k)$ $k<<N$

Latent Semantic Analysis (LSA)

- $D=USV^T$

$d=(w_1, \ldots, w_N)$

- $D'=US_kV^T$

$d=(z_1,\ldots,z_k)$ $k<<N$

- $\min \| D - D' \|^2_F = \sum (d[i][j]-d'[i][j])^2$

(Doc 1)	*G: Do you see the lake with the black swan?*	*Query–yn*
(Doc 2)	*F: Yes, I do*	*Reply–y*
(Doc 3)	*G: Ok,*	*Ready*
(Doc 4)	*G: draw a line straight to it*	*Instruct*
(Doc 5)	*F: straight to the lake?*	*Check*
(Doc 6)	*G: yes, that's right*	*Reply–y*
(Doc 7)	*F: Ok, I'll do it*	*Acknowledge*

Figure 1: A hypothetical dialogue annotated with MapTask tags

	(Doc 1)	(Doc 2)	(Doc 3)	(Doc 4)	(Doc 5)	(Doc 6)	(Doc 7)
do	1	1	0	0	0	0	1
see	1	0	0	0	0	0	0
lake	1	0	0	0	1	0	0
black	1	0	0	0	0	0	0
swan	1	0	0	0	0	0	0
yes	0	1	0	0	0	1	0
ok	0	0	1	0	0	0	1
draw	0	0	0	1	0	0	0
line	0	0	0	1	0	0	0
straight	0	0	0	1	1	0	0
to	0	0	0	1	1	0	0
it	0	0	0	1	0	0	1
that	0	0	0	0	0	1	0
right	0	0	0	0	0	1	0

Table 1: The 14-dimensional word-document matrix W

LSA uses co-occurrence statistics

	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10
t1	1	2	2	7	9	0	2	0	0	0
t2	7	8	9	6	0	0	0	1	0	0
t3	3	1	2	1	8	0	0	0	0	0
t4	8	0	4	0	6	0	1	0	0	0
t5	0	2	2	0	5	0	0	0	0	0
t6	5	9	6	2	2	0	0	0	0	0
t7	9	0	1	8	8	0	0	0	0	0
t8	0	0	0	2	0	9	2	5	5	9
t9	1	0	0	0	0	6	1	2	1	7
t10	0	0	0	0	0	2	0	3	5	1
t11	0	0	0	0	0	6	8	0	3	8
t12	0	0	1	0	0	6	1	8	6	1
t13	0	0	0	0	0	1	0	8	8	6
t14	0	0	0	0	2	0	7	2	5	6

$D'=US_kV^T$

	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10
t1	5	3	3	3	4	0	0	0	0	0
t2	6	4	4	5	6	0	0	0	0	0
t3	3	2	2	2	3	-1	0	0	-1	-1
t4	4	3	3	3	4	-1	0	0	-1	0
t5	2	1	1	1	2	-1	0	0	-1	-1
t6	5	3	3	4	5	-1	0	-1	-1	-1
t7	6	4	5	5	6	-1	0	0	-1	-1
t8	0	0	0	0	0	6	3	5	6	8
t9	0	0	0	0	0	3	2	3	3	4
t10	-1	-1	-1	0	-1	2	1	2	2	2
t11	-1	-1	0	0	0	5	3	4	5	6
t12	0	-1	0	0	0	4	2	3	4	5
t13	-1	-1	-1	0	0	4	2	4	5	6
t14	0	0	0	0	0	3	2	3	3	4

Feature LSA (FLSA)

- Dialog acts are treated as documents

Feature LSA (FLSA)

- Dialog acts are treated as documents
- Compute LSA representations for DA's

Feature LSA (FLSA)

- Dialog acts are treated as documents
- Compute LSA representations for DA's
- Use features other than terms in the DA vectors:
  - POS, syntactic information
  - previous DA, game

Feature LSA (FLSA)

- Dialog acts are treated as documents
- Compute LSA representations for DA's
- Use features other than terms in the DA vectors:
  - POS, syntactic information
  - previous DA, game
- Compute LSA on the DA vectors extended with new features - FLSA

Corpus 1: CallHome Spanish

- 120 telephone conversations in Spanish (family, friends)
  - 12066 unique words, 44628 DA's
  - 232 tags – unified in 37, 10, 8 groups

Corpus 1: CallHome Spanish

- 120 telephone conversations in Spanish (family, friends)
  - 12066 unique words, 44628 DA's
  - 232 tags – unified in 37, 10, 8 groups
- Tags:
  - DA (statement, question, answer...)
  - Move (initiative, response, feedback)
  - Game (information, directive)
  - Activities (gossip, argue)

Corpus 2: MapTask

- 128 dialogs, map task experiment
  - 1835 unique words, 27084 DA's
- Tags:
  - DA's (=moves) (instruct, explain,...)
  - Games (clarification, ...)
  - Transaction (normal, review, overview, irrelevant)

Corpus 3: DIAG-NLP

- Computer mediated tutoring dialogs between a tutor and a student
  - 23 dialogs
  - 670 unique words, 660 DA's

Corpus 3: DIAG-NLP

- Computer mediated tutoring dialogs between a tutor and a student
  - 23 dialogs
  - 670 unique words, 660 DA's
- Tags:
  - 4 DA's (problem solving, judgment, domain knowledge, other)
  - Consult Type (type of student query)

New Features

- POS, SRule (declarative, Wh-question)
- Duration
- Speaker (MapTask: Giver, Follower)
- Previous DA
- Game
- Initiative
- Combination

Performance Comparison

Corpus	Baseline	LSA	FLSA	Best other
CallHome37	42.68%	65.36%	74.87%	76.20%
CallHome10	42.68%	68.91%	78.88%	76.20%
MapTask	20.69%	42.77%	73.91%	62.10%
DIAG-NLP	43.64%	75.73%	74.81%	n.a.

Baseline is picking the most frequent DA in each corpus
LSA, FLSA – classification using the training DA vectors

Features Contribution

- Features that did not help
  - POS
  - SRule
  - Previous DA

Features Contribution

- Features that did not help
  - POS
  - SRule
  - Previous DA
- Features that helped
  - Game
  - Speaker
  - Initiative
  - Combinations of these

MapTask

LSA 42.77%

MapTask 41.84% SRule

MapTask 43.28% POS

MapTask 43.59% Duration

MapTask 46.91% Speaker

MapTask 47.09% Previous DA

MapTask 66.00% Game

MapTask 69.37% Game+Prev. DA

MapTask 73.25% Game+Speaker+Prev. DA

MapTask 73.91% Game+Speaker

Comments

- Not clear how to interpret LSA in this setting:
  - classification is done by finding the most similar training DA. LSA accounts for semantic similarity.
  - only works withing the same dataset?

Comments

- Not clear how to interpret LSA in this setting:
  - classification is done by finding the most similar training DA. LSA accounts for semantic similarity.
  - only works withing the same dataset?
- Features are controversial because the labels are not known for new data

Comments

- Not clear how to interpret LSA in this setting:
  - classification is done by finding the most similar training DA. LSA accounts for semantic similarity.
  - only works withing the same dataset?
- Features are controversial because the labels are not known for new data
- “Game” contains a lot of information about the DA's label
- Previous DA can be inferred by the system, but this feature did not help

SVMs and HMMs for DA Tagging

Recognizing Maptask Acts

- Assume:
  - Word-level transcription
  - Segmentation into utterances,
  - Ground truth DA tags
- Goal: Train classifier for DA tagging
  - Exploit:
    - Lexical and prosodic cues
    - Sequential dependencies b/t Das
  - 14810 utts, 13 classes

Features for Classification

- Acoustic-Prosodic Features:
  - Pitch, Energy, Duration, Speaking rate
    - Raw and normalized, whole utterance, last 300ms
    - 50 real-valued features

Features for Classification

- Acoustic-Prosodic Features:
  - Pitch, Energy, Duration, Speaking rate
    - Raw and normalized, whole utterance, last 300ms
    - 50 real-valued features
- Text Features:
  - Count of Unigram, bi-gram, tri-grams
    - Appear multiple times
    - 10000 features, sparse

Classification with SVMs

- Support Vector Machines

Classification with SVMs

- Support Vector Machines

  - Create n(n-1)/2 binary classifiers
    - Weight classes by inverse frequency
    - Learn weight vector and bias, classify by sign

Classification with SVMs

- Support Vector Machines

  - Create n(n-1)/2 binary classifiers
    - Weight classes by inverse frequency
    - Learn weight vector and bias, classify by sign

  - Platt scaling to convert outputs to probabilities

Incorporating Sequential Constraints

- Some sequences of DA tags more likely:

Incorporating Sequential Constraints

- Some sequences of DA tags more likely:
  - E.g. P(affirmative after y-n-Q) = 0.5
  - P(affirmative after other) = 0.05

Incorporating Sequential Constraints

- Some sequences of DA tags more likely:
  - E.g. P(affirmative after y-n-Q) = 0.5
  - P(affirmative after other) = 0.05
- Learn $P(yi|yi\text{-}1)$ from corpus
  - Tag sequence probabilities
  - Platt-scaled SVM outputs are $P(y|x)$

Incorporating Sequential Constraints

- Some sequences of DA tags more likely:
  - E.g. P(affirmative after y-n-Q) = 0.5
  - P(affirmative after other) = 0.05
- Learn P(yi|yi-1) from corpus
  - Tag sequence probabilities
  - Platt-scaled SVM outputs are P(y|x)
- Viterbi decoding to find optimal sequence

Results

	SVM Only	SVM+Seq
Text Only	58.1	59.1
Prosody Only	41.4	42.5
Text+Prosody	61.8	65.5

Observations

- DA classification can work on open domain
  - Exploits word model, DA context, prosody
  - Best results for prosody+words
  - Words are quite effective alone – even ASR
- Questions:

Observations

- DA classification can work on open domain
  - Exploits word model, DA context, prosody
  - Best results for prosody+words
  - Words are quite effective alone – even ASR
- Questions:
  - Whole utterance models? – more fine-grained
  - Longer structure, long term features