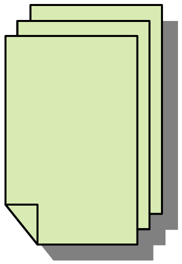
Widening the NLP pipeline for Spoken Language Processing

Srinivas Bangalore, AT&T-Research

Outline

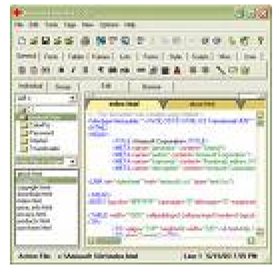
- Natural Language Processing Pipeline
  - Text input
  - Speech input
- Uniform decoding framework
- Case Studies
  - Call-type classification
  - Speech translation
  - Multimodal language processing

Text-based Natural Language Processing

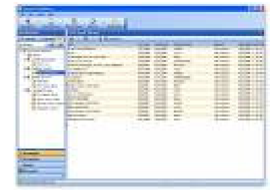


Text Document(s)

Web Documents



Email Messages



Blogs

Input-specific Normalization → Text → NLP Pipeline → Richly Annotated Text → Task-specific Transduction

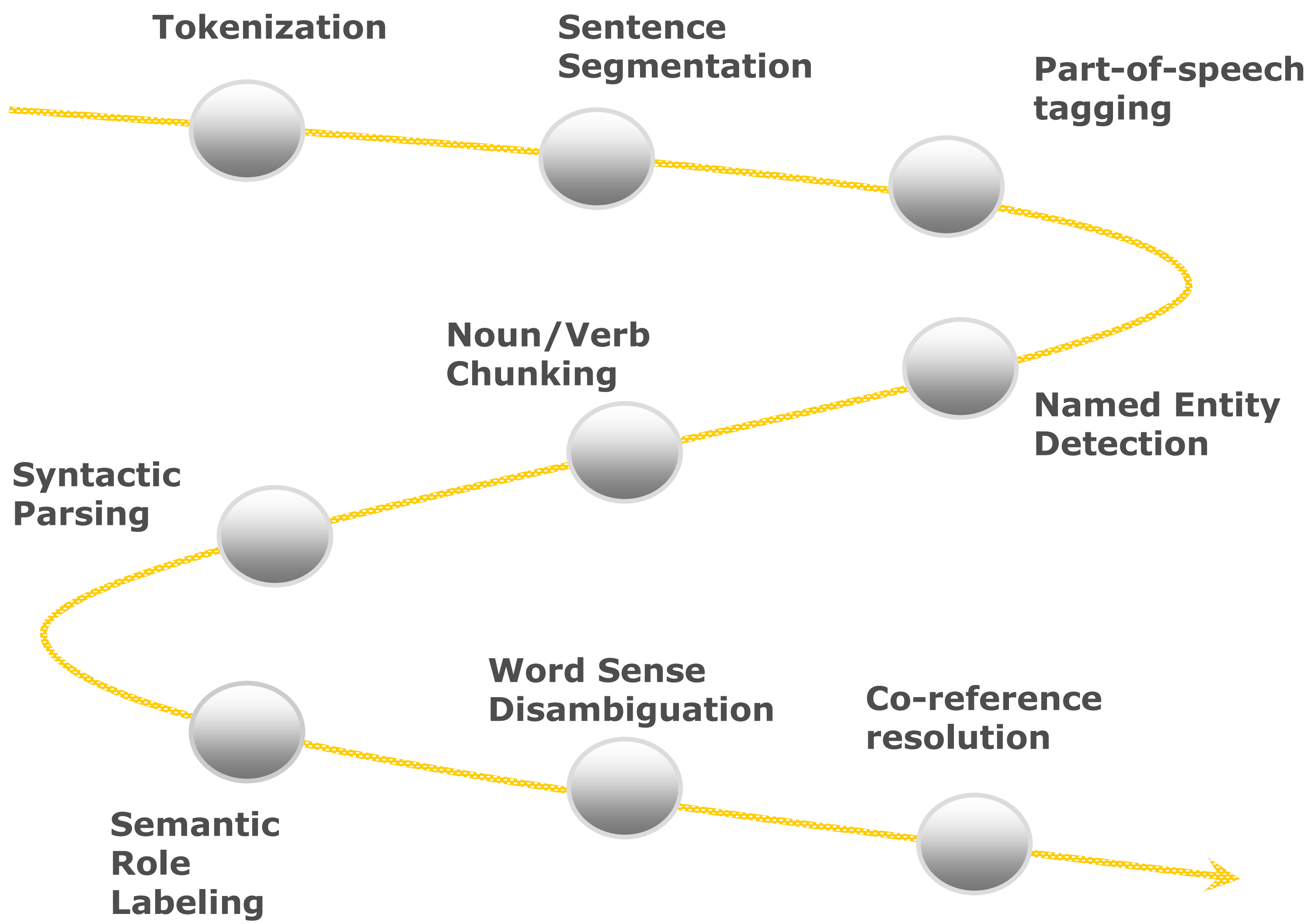
Summarization

Machine Translation

Question Answering

Information Extraction

NLP Pipeline: Beads on a String

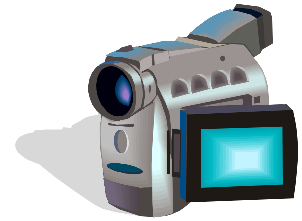


Spoken Language Processing

Telephone conversations

News Broadcasts

Voicemail Messages



Movies

Speech Recognition → ASR output → NLP Pipeline → Richly Annotated Text → Task-specific Transduction

Summarization

Machine Translation

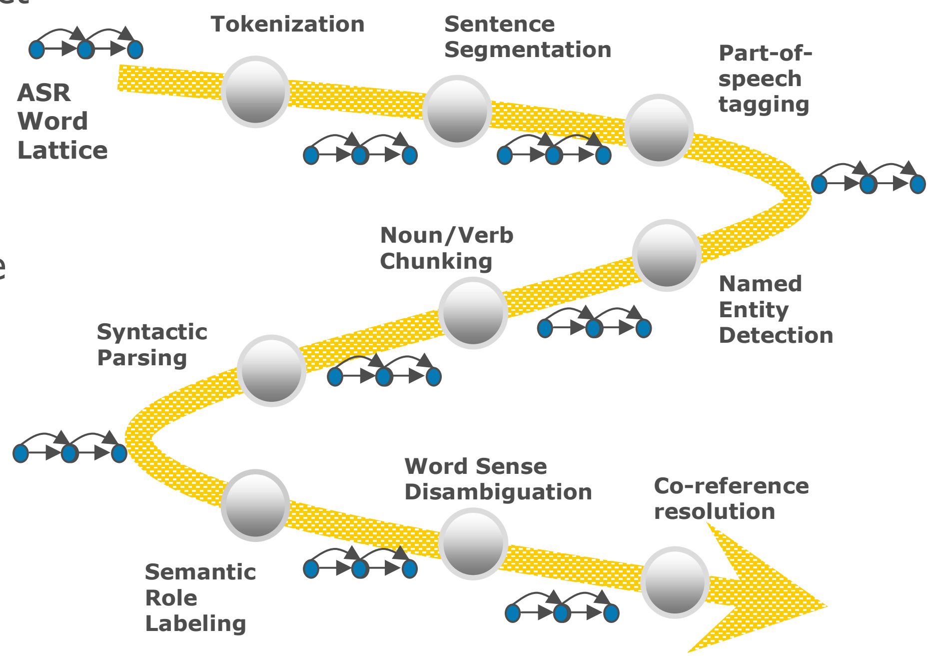
Question Answering

Information Extraction

Interactive Dialog

Widening the NLP pipeline

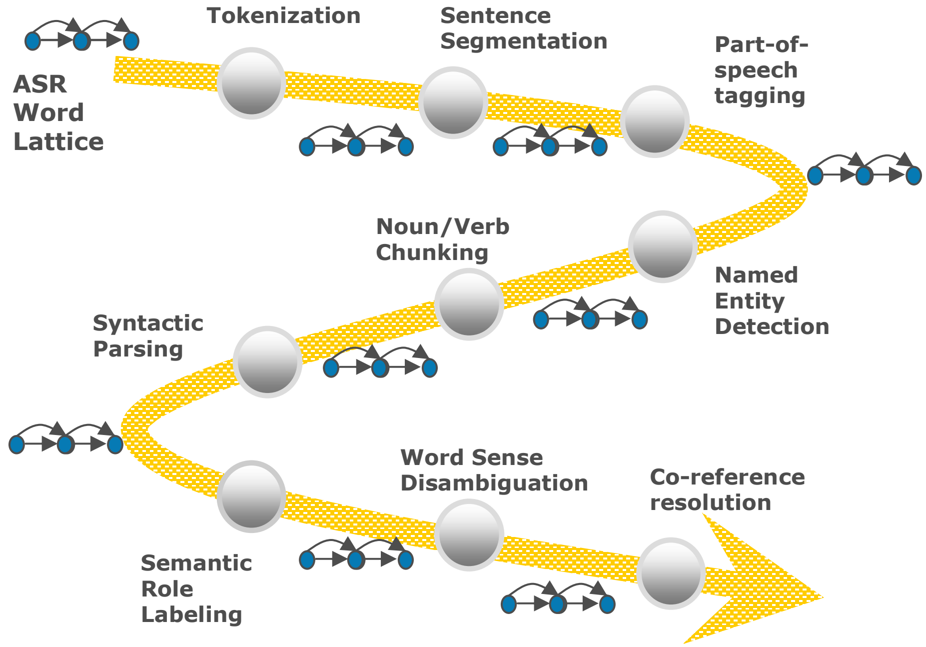
- Passing one-best solution is sub-optimal.
- **Error in processing models**
  - Most modules in the pipeline are not perfect
  - Error propagation down the pipeline
- **Ambiguity in NLP**
  - *"John saw a man with a telescope"*
  - Postpone ambiguity resolution down the pipeline
  - Until information is available to resolve the ambiguity
- **N-best solutions**
  - List of solutions ranked by some goodness criteria
- **Weighted packed representations**
  - Lattices for linear outputs
  - Forests for hierarchical outputs



- **N-best versus Lattices/Forests**
  - N needs to be very large for substantially different solution
  - Repeated computation is factored out
    - Significant parts are shared across n-best solutions

A word about decoders

- Specialized decoder for each task
  - Use weighted lattices as input
  - Produce weighted lattices as output
- Uniform decoding framework
  - Most NL processing steps can be encoded as token tagging tasks.
  - … *word*/?? …
  - Approximation for other steps
    - Attachment in parsing
- Weighted finite-state transducers



Weighted Finite-State Transducers (WFST)

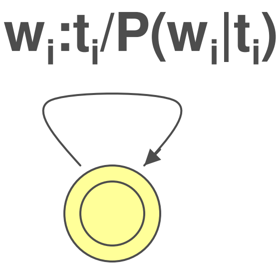
- Provide efficient ways of representing weighted ambiguous hypotheses.
- Closed under composition
  - straightforward integration of finite-state constraints.
  - allows for modular development without loss of optimality of the solution.
- Decoding: linear in the input size.
- Multi-tape finite-state automata used to represent constraints from different levels of language processing.
- Extensively used for speech and language processing.

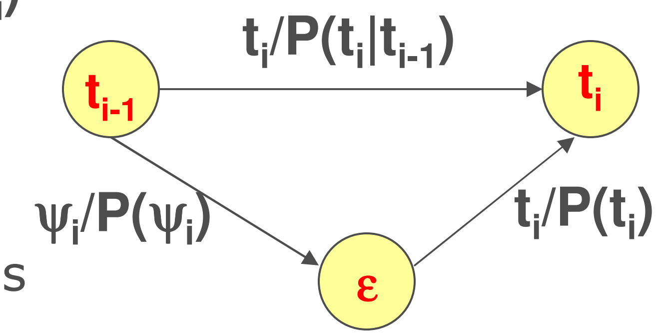
Decoders as WFSTs

- Grammar based decoding models
  - Regular expressions (e.g. dates, telephone numbers, name lists)
  - Context-free grammars (syntactic parsers)
    - Approximation techniques (Nederhof 1997, Pereira and Wright 1997)
- HMM-based generative model

$$T^* = \operatorname{argmax} \prod^{n} P(w_i \mid t_i) * P(t_i \mid t_{i-1})$$

  - (Schabes and Roche 1997)





- Discriminatively trained classification models

$$T^* = \operatorname{argmax} \prod^{n} P(t_i \mid f(t^{1}_{i-1}, w^{1}_{n}))$$

  - Decision Trees to FSTs (Sproat and Riley, 1996); Adaboost to FSTs (Bangalore, 2004)
  - Encode features and weights as context-dependent rewrite rules (CDR)

$$\phi \rightarrow \psi \mid \gamma \text{ --- } \delta$$

  - Compile CDRs into FSTs (Johnson 1972, Kaplan and Kay 1994, Mohri and Sproat 1996)

Outline

- Natural Language Processing Pipeline
  - Text input
  - Speech input
- Uniform decoding framework
- Case Studies
  - Call-type classification
  - Speech translation
  - Multimodal language processing

Call-type classification

Calls are classified based on user's response to an opening prompt.

– "How may I help you" (Gorin et.al. 1997); BBN call director (Natarajan et.al. 2002); (Chelba and Acero 2003); (Cox 2003)

Training data:

I would like to speak to an operator : Request(customer_care)
What is my account balance: Request(account_balance)
I'd like to have a copy of my March bill: Request(copy_bill)
How do I pay my bill: Ask(bill_payment)

Classification model:

$$topclass = \arg\max_{class} P(class \mid Ngrams(input))$$

ASR output is classified

– one-best, n-best, word lattice

Call-type classification error rates Results from (Haffner 2005)

Top class error rate after rejecting 30% low confidence examples:

- -3 inputs: 1-best sentence, 10 best, full lattice
- - trigram word features

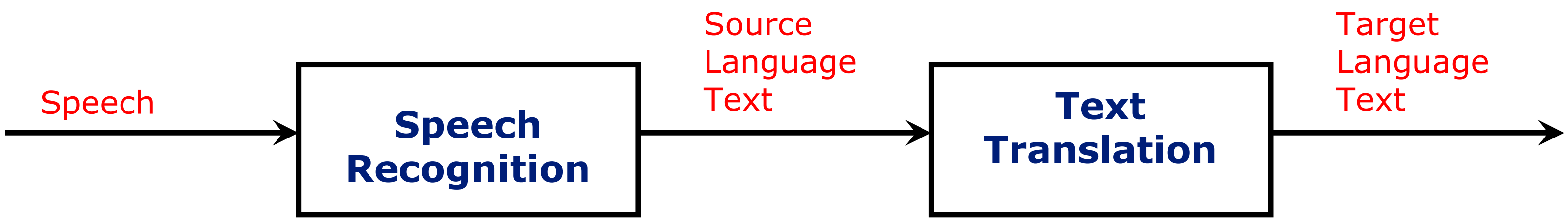
ASR word accuracy about 70% for the three applications

Classifier	Input	App1 (82 classes)	App2 (97 classes)	App3 (64 classes)
Poly2 SVM	**1-best**	**12.9**	**8.44**	**4.66**
Poly2 SVM	10-best	11.3	7.45	4.37
Poly2 SVM	**Lattice**	**10.2**	**6.68**	**3.37**

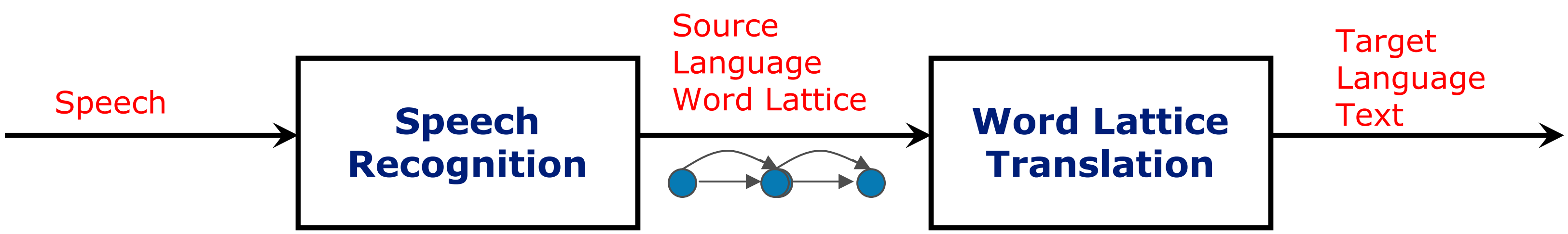
Classification of ASR word lattices consistently outperforms classification of one-best ASR output.

Spoken Language Translation

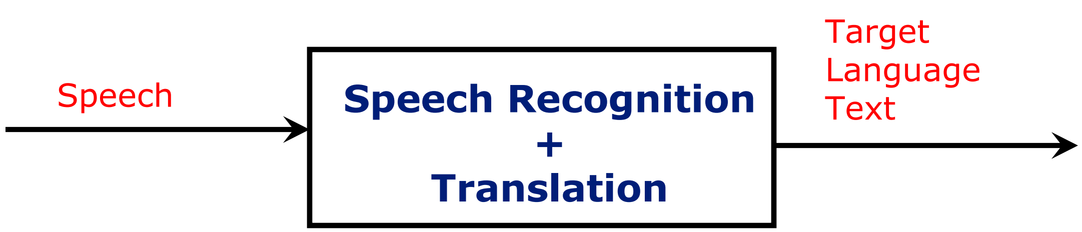
- Two-step process (narrow pipe)



- Two-step process (broader pipe)



- Tight-coupling (integrated ASR+MT)



FST-based Spoken Language Translation

- Finite-state transducer based spoken language translation
  - Lexical choice and reordering are modeled using finite-state transducers
  - Vidal et al 1997, Ney 1999, Bangalore and Riccardi 2000, Zhou et al 2005, Shankar and Byrne 2005, Crego 2004.
  - *T* estimated from bilingual phrases/tuples, source *F*, decoded target E*

$$E_{lex} = \pi_1(best(F \circ T))$$

$$E^* = best(permute(E_{lex}) \circ LM_E)$$

- FST-based Eutrans II Italian-English task (Matusov, Kanthak, Ney ICASSP 2006)
  - 23.7% ASR word-error rate

Method	WER(%)	BLEU
One-best ASR output	37.4	51.3
Word-lattices ASR output	36.6	52.4
ASR+MT integrated	36.3	52.6

Multimodal Language Processing

Multimodal interfaces: allow for multiple modes of input

- Pen/hand gestures, handwriting and speech

Interpretation of input

- derived by fusing information distributed in multiple input modalities
- Bolt 1980, Cohen et al 1997, Johnston and Bangalore 2000, Johnston et al 2002, Joyce 2004, Meng et al 2006

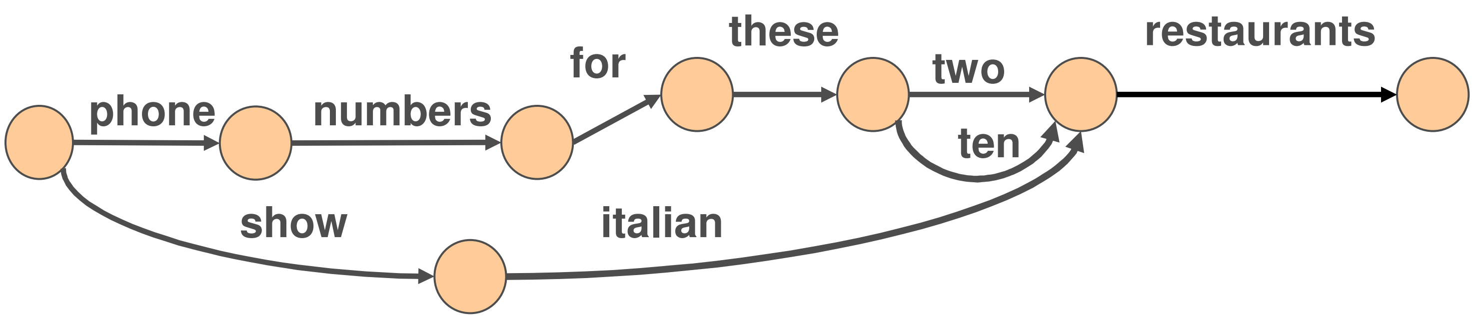
Challenges:

- Interpretation ambiguity
  - Each combination of strokes as a candidate for handwriting and gesture recognition
  - Even simple inputs can have highly ambiguous interpretations

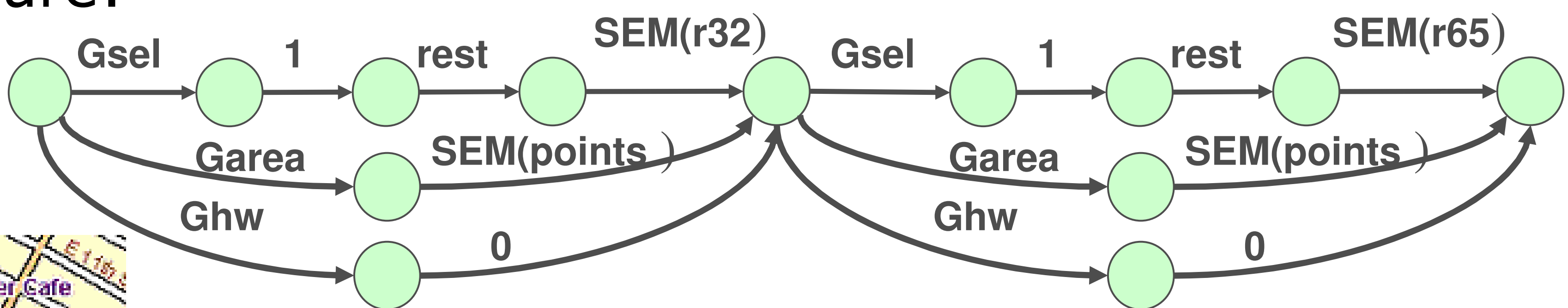
- Speech and gesture recognition errors
- Modality Synchronization
  - Alignment between input lattices

Representation of input and output streams

Speech:

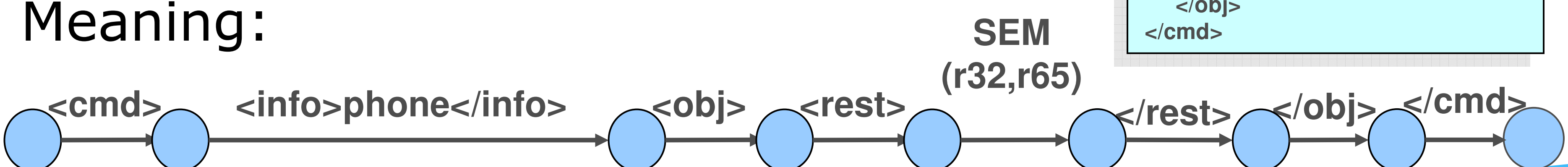


Gesture:




```
<cmd>
  <info>phone</info>
   <obj>
      <rest>r32,r65</rest>
   </obj>
</cmd>
```

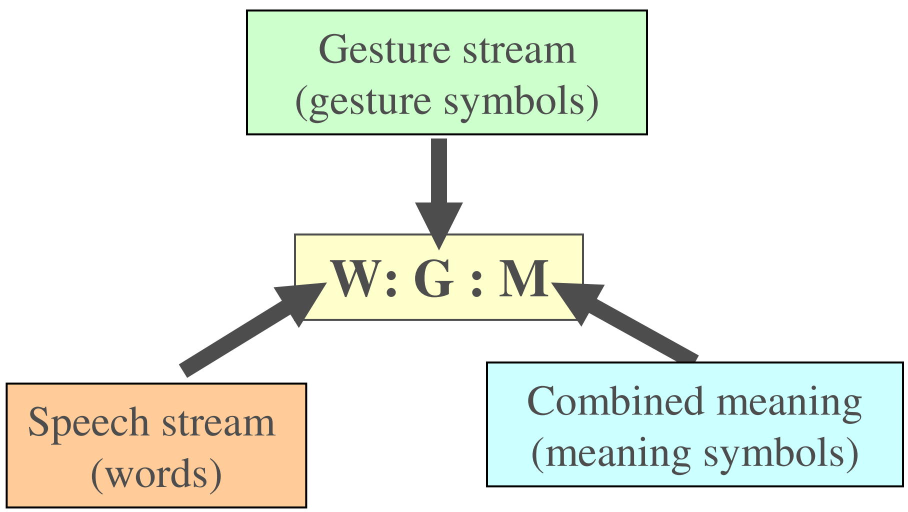
Meaning:



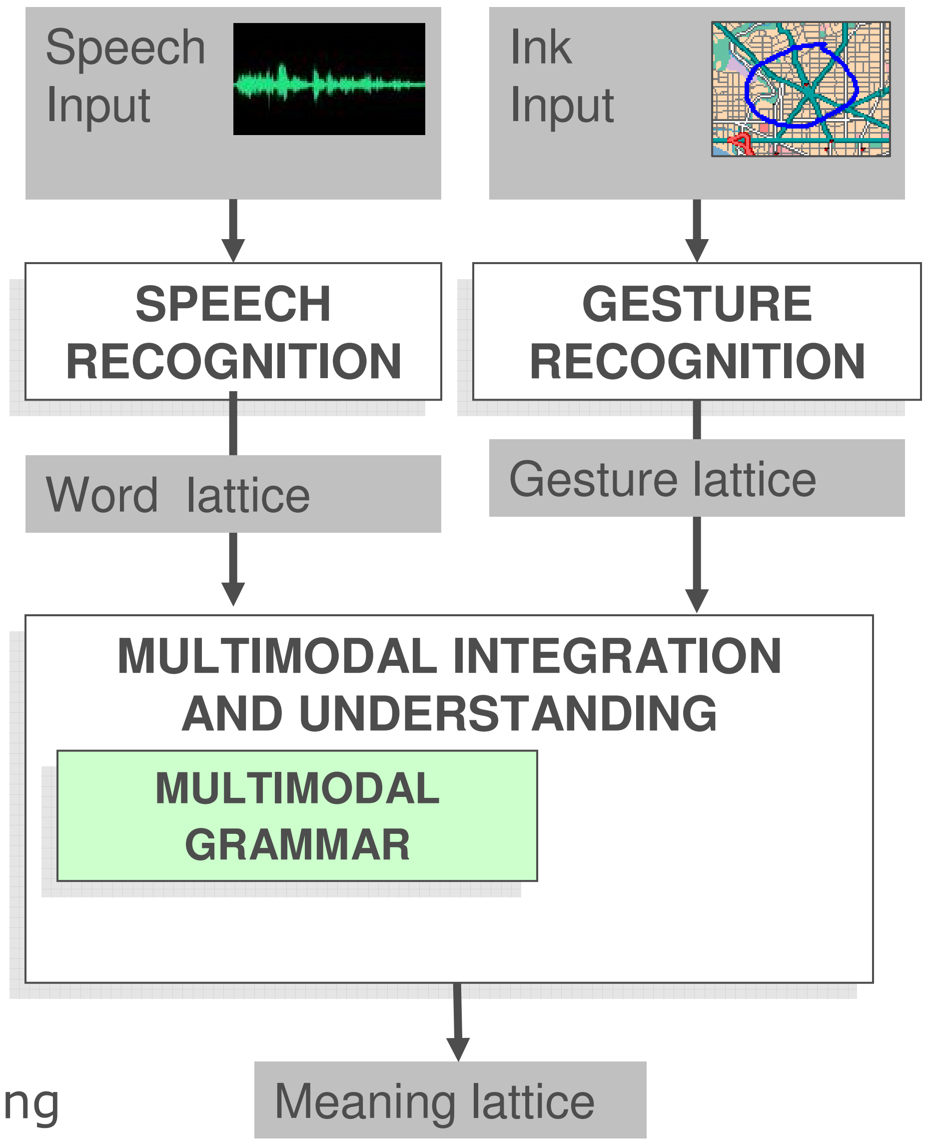
Multimodal grammars

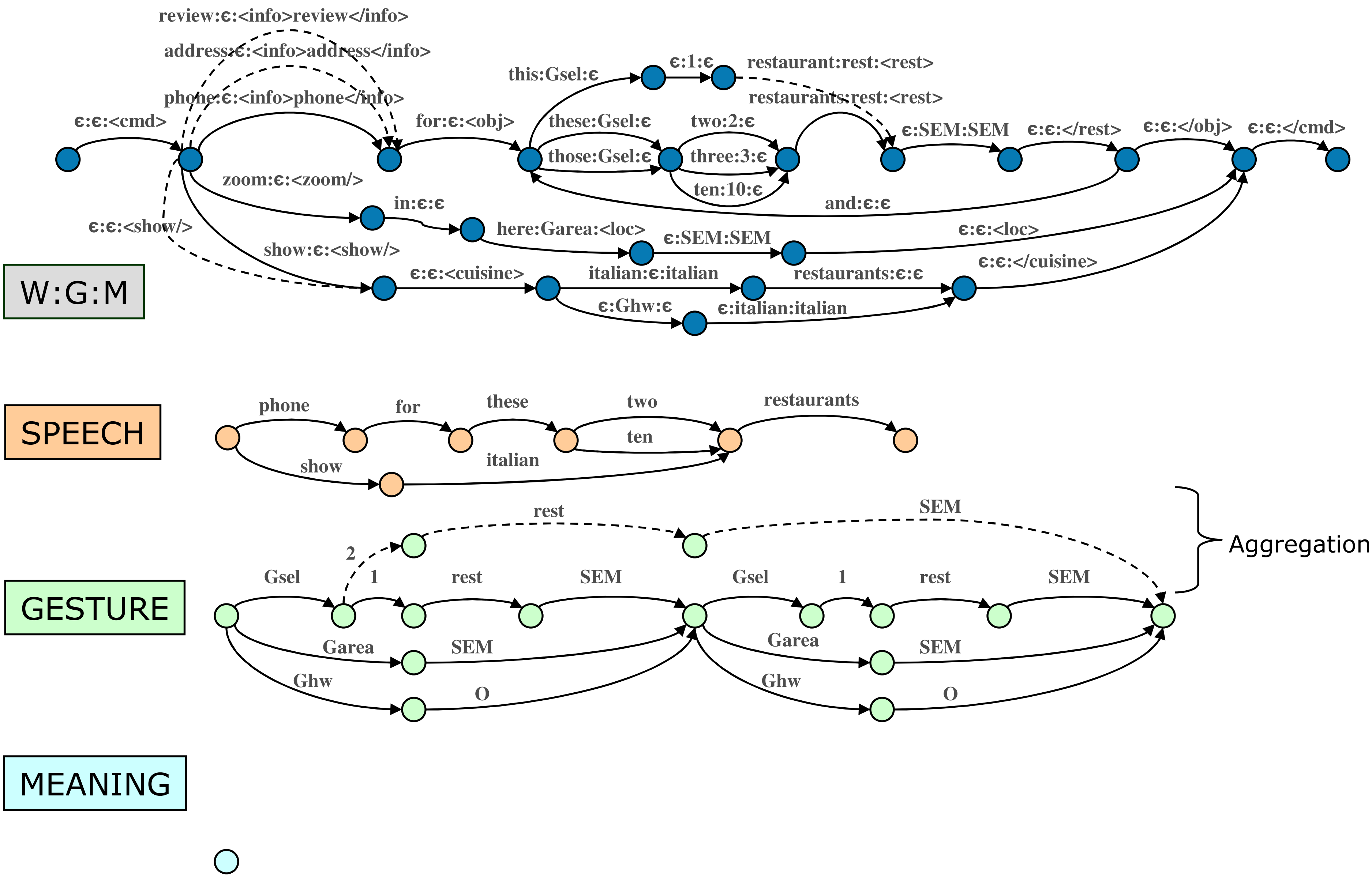
Multimodal context-free grammar

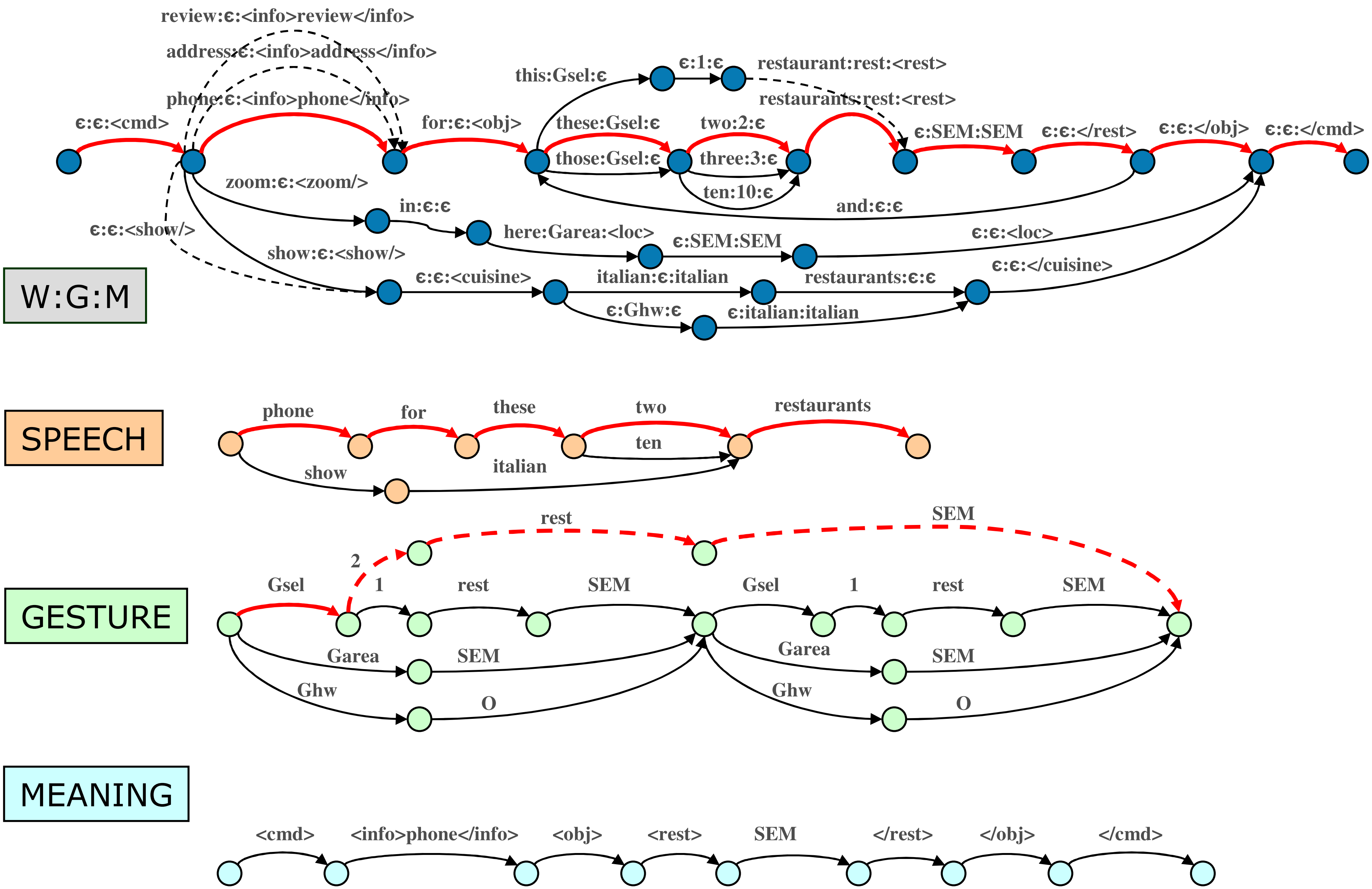
- Terminals are multimodal tokens consisting of three components:



- Grammar rules encode
  - Gesture and speech alignment
  - Gesture-speech combined meaning




W:G:M
review:ε:<info>review</info>
address:ε:<info>address</info>
phone:ε:<info>phone</info>
ε:ε:<cmd>
for:ε:<obj>
this:Gsel:ε
ε:1:ε
restaurant:rest:<rest>
restaurants:rest:<rest>
these:Gsel:ε
two:2:ε
those:Gsel:ε
three:3:ε
ten:10:ε
ε:SEM:SEM
ε:ε:</rest>
ε:ε:</obj>
ε:ε:</cmd>
zoom:ε:<zoom/>
in:ε:ε
and:ε:ε
ε:ε:<show/>
here:Garea:<loc>
ε:SEM:SEM
ε:ε:<loc>
show:ε:<show/>
ε:ε:<cuisine>
italian:ε:italian
restaurants:ε:ε
ε:ε:</cuisine>
ε:Ghw:ε
ε:italian:italian
SPEECH
phone
for
these
two
ten
restaurants
show
italian
GESTURE
rest
SEM
2
Gsel
1
rest
SEM
Gsel
1
rest
SEM
Garea
SEM
Garea
SEM
Ghw
O
Ghw
O
Aggregation
MEANING


review:ε:<info>review</info>
address:ε:<info>address</info>
phone:ε:<info>phone</info>
ε:ε:<cmd>
for:ε:<obj>
this:Gsel:ε
ε:1:ε
restaurant:rest:<rest>
restaurants:rest:<rest>
these:Gsel:ε
two:2:ε
ε:SEM:SEM
ε:ε:</rest>
ε:ε:</obj>
ε:ε:</cmd>
those:Gsel:ε
three:3:ε
ten:10:ε
zoom:ε:<zoom/>
in:ε:ε
and:ε:ε
ε:ε:<show/>
here:Garea:<loc>
ε:SEM:SEM
ε:ε:<loc>
show:ε:<show/>
ε:ε:<cuisine>
italian:ε:italian
restaurants:ε:ε
ε:ε:</cuisine>
ε:Ghw:ε
ε:italian:italian
W:G:M
SPEECH
phone
for
these
two
restaurants
ten
show
italian
GESTURE
rest
SEM
2
Gsel
1
rest
SEM
Gsel
1
rest
SEM
Garea
SEM
Garea
SEM
Ghw
O
Ghw
O
MEANING
<cmd>
<info>phone</info>
<obj>
<rest>
SEM
</rest>
</obj>
</cmd>

Finite-state edit machines: Robust interpretation

- Transform ASR output so that it can be assigned a meaning by the FST-based Multimodal Understanding model.

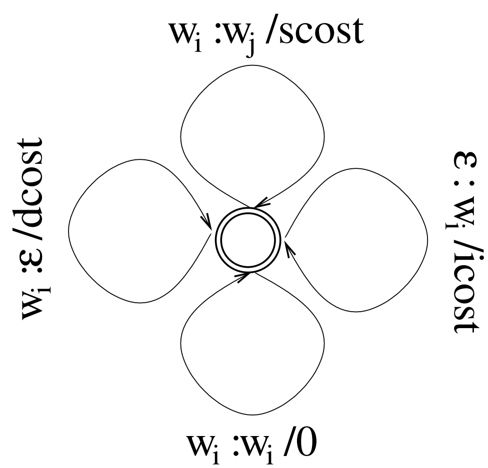
phone ***restaurant*** *these* *to* *restaurants* *please*

phone ***for*** *ε* *these* ***two*** *restaurants* *ε*

- Decoding: $$s^* = \arg\min_{s} \lambda_s \circ \lambda_{Edit} \circ \lambda_{Grammar}$$

- MATCH domain concept sentence accuracy
  (Bangalore and Johnston 2006)

	Concept Sentence Accuracy
No Edit	38.9
One-best Edit	60.2
Lattice Edit	63.2



Edit machine (insert, substitute, delete, identity arcs).

Summary

- Widening the NLP pipeline for spoken language processing
  - Imperfect output from speech recognition and other processing components
  - Inherent ambiguity in language
  - N-best or lattice representations
- Extending decoders to cope with lattice input
  - FST as a uniform decoding framework
  - Grammar-based, HMM-based, Classification-based decoders
- Case Studies:
  - Call-type classification
  - Spoken Language Translation
  - Multimodal Language Processing
- Issues:
  - Combining weights across multiple disambiguation models
  - Search and prune during FST composition (Lazy evaluation)