# Lecture 12: Audio Databases

1. **ASR wrap-up**
2. **Spoken document retrieval**
3. **General audio databases**
4. **Open issues**

Dan Ellis <dpwe@ee.columbia.edu>
http://www.ee.columbia.edu/~dpwe/e6820/

# Outline

2 **ASR wrap-up**
- Decoding
- Discriminant modeling
- Adaptation
- Confidence measures

2 **Spoken document retrieval**

3 **General audio databases**

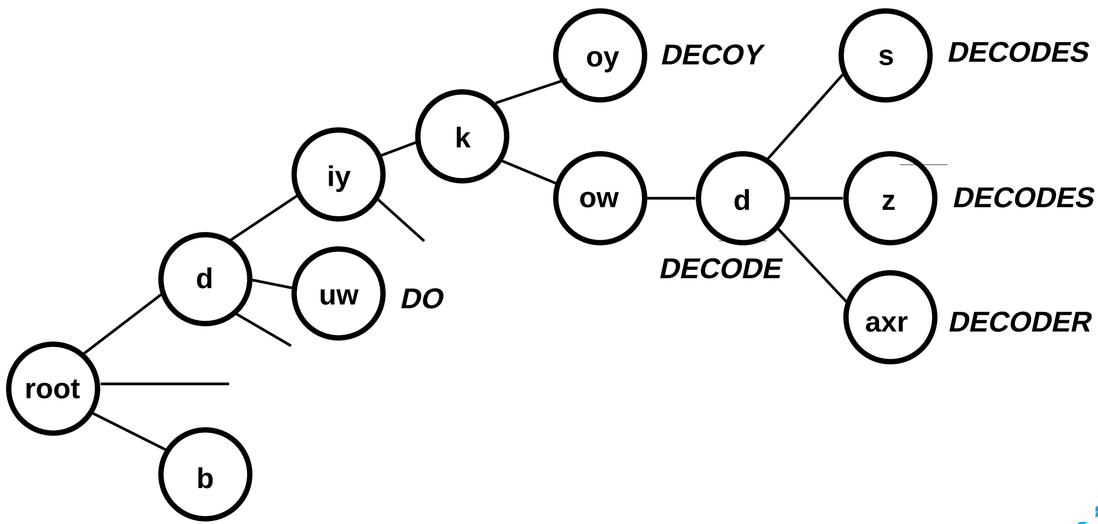
4 **Open issues**

1

# ASR wrap-up: Decoding

- **How to find the MAP word sequence?**
- **States, prons, words define one big HMM**
  - with 100,000+ individual states for LVCSR!

→ **Exploit hierarchic structure**
  - phone states independent of word
  - next word (semi) independent of word history



# Decoder pruning

- **Searching 'all possible word sequences'?**
  - need to restrict search to most promising ones: *beam search*
  - sort by estimates of total probability
    = Pr(so far) + lower bound estimate of remains
  - trade search errors for speed

- **Start-synchronous algorithm:**
  - extract top hypothesis from queue:

$$[P_n, \{w_1, \ldots, w_k\}, n]$$

*pr. so far* *words* *next time frame*

  - find plausible words $\{w^i\}$ starting at time $n$
    $\rightarrow$ new hypotheses:

$$[P_n \cdot p(X_n^{n+N-1} | w^i) \cdot p(w^i | w_k \ldots), \{w_1, \ldots, w_k, w^i\}, n + N]$$

  - discard if too unlikely, or queue is too long
  - else re-insert into queue and repeat

# Discriminant models

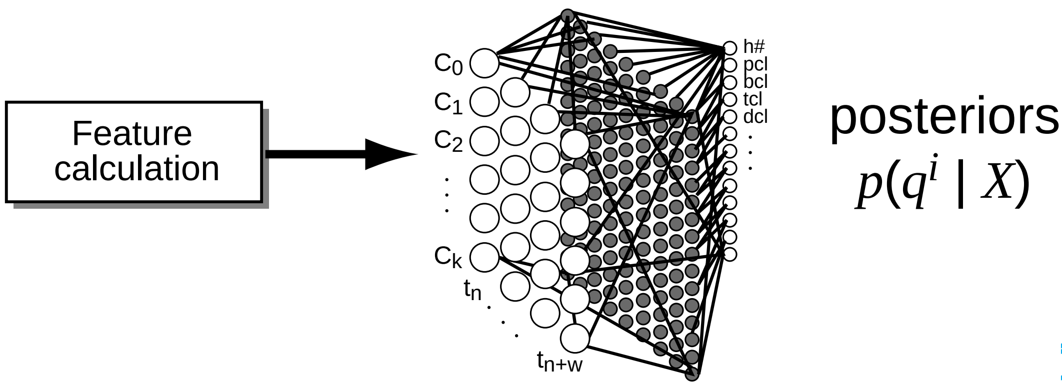
- **EM training of HMMs is *maximum likelihood***
  - i.e. local $\max p(X_{trn} \mid \Theta)$
  - converges to Bayes optimum ... in the limit
- **Decision rule is** $\max p(X \mid M) \cdot p(M)$
  - training will increase $p(X \mid M_{correct})$
  - may also increase $p(X \mid M_{wrong})$ ...more?
- **Discriminant training tries directly to increase discrimination between right & wrong models**
  - e.g. Maximum Mutual Information (MMI)

$$I(M_j, X \mid \Theta) = \log \frac{p(M_j, X \mid \Theta)}{p(M_j \mid \Theta)\, p(X \mid \Theta)}$$

$$= \log \frac{p(X \mid M_j, \Theta)}{\sum p(X \mid M_k, \Theta)\, p(M_k \mid \Theta)}$$

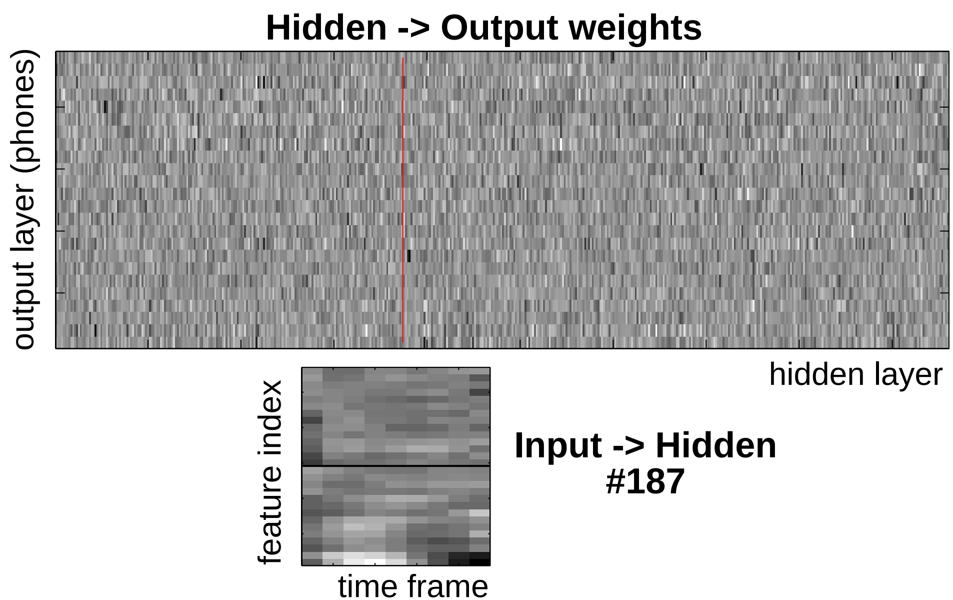
# Neural Network Acoustic Models

- **Single model generates posteriors directly for all classes at once = *frame*-discriminant**
- **Use regular HMM *decoder* for recognition**
  - set $b_i(x_n) = p(x_n | q^i) \propto p(q^i | x_n) / p(q^i)$
- **Nets are less sensitive to input representation**
  - skewed feature distributions
  - correlated features
- **Can use temporal context window to let net 'see' feature dynamics:**

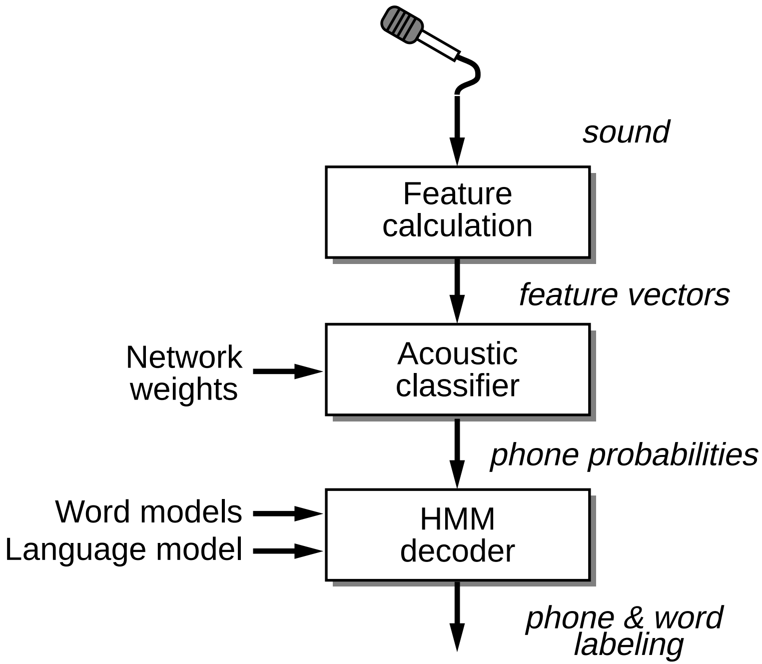


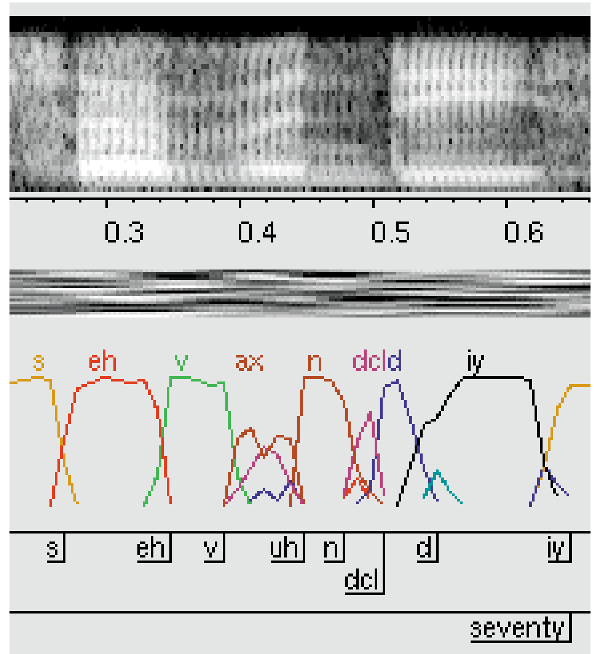
# Neural nets: Practicalities

- **Typical net sizes:**
  - input layer: 9 frames x 9-40 features ~ 300 units
  - hidden layer: 100-8000 units, dep. train set size
  - output layer: 30-60 context-independent phones
- **Hard to make context dependent**
  - problems training many classes that are similar?
- **Representation is opaque:**



# Recap: Recognizer Structure



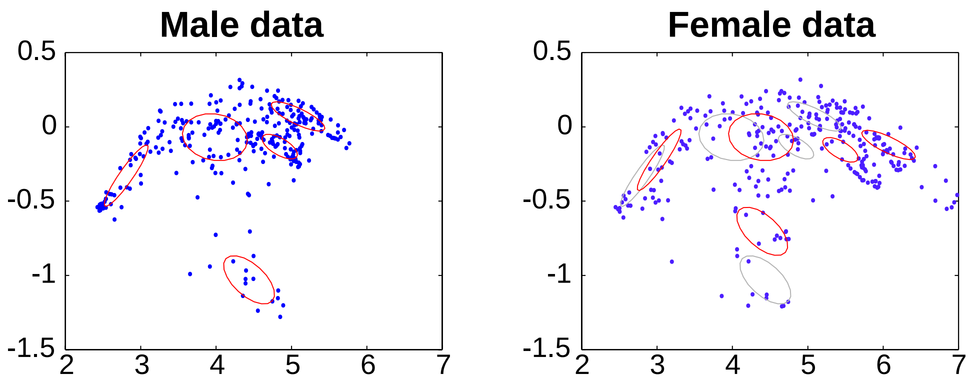

- **Now we have it all!**

# Model adaptation

- **Practical systems often suffer from *mismatch***
  - test conditions are not like training data: accent, microphone, background noise ...
- **Desirable to continue tuning during recognition = adaptation**
  - but: no ‘ground truth’ labels or transcription
- **Assume that recognizer output is correct; Estimate parameters from those labels**
  - like Viterbi training
  - can iterate until convergence
- **Normally have little adaptation data**
  - want to adapt using data from one speaker/ condition only (clustering)
  - hence, can estimate only a few parameters (not update whole acoustic model)

# Maximum-Likelihood Linear Regression

- **Approximate mismatch as an 'affine transform':**




- **Estimate matrix to transform means of GMM components**

$$\hat{\mu}_j = A \cdot \mu_j + b$$

  - find maximum likelihood solution via EM over adaptation data
  - ML only works for transforming *model*, not data

- **Typically 10-20% relative WER reduction**
  - given enough data from one condition

# Recognizer outputs

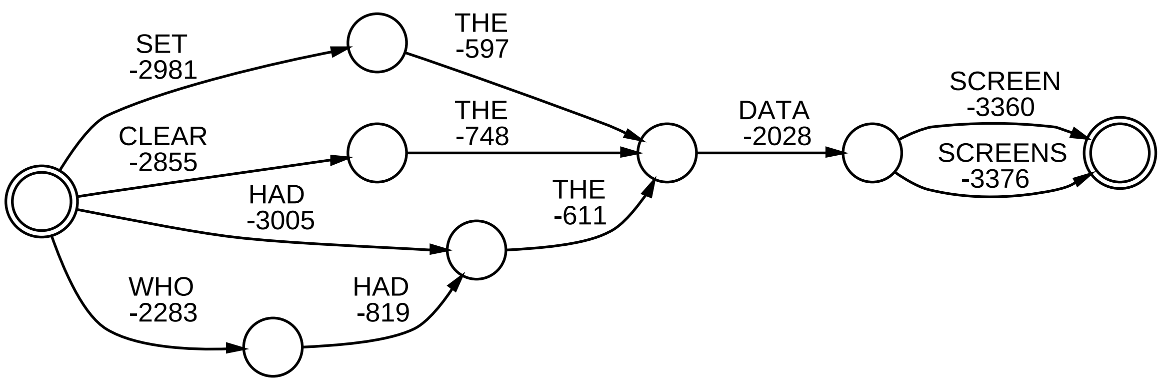
- **Basic output of decoder is most probable word sequence**

```
SET THE DATA SCREEN
```

- **Other possible outputs:**
  - n best word sequences (with likelihoods)

```
-9076  SET THE DATA SCREEN
-9092  SET THE DATA SCREENS
-9158  CLEAR THE DATA SCREEN
```

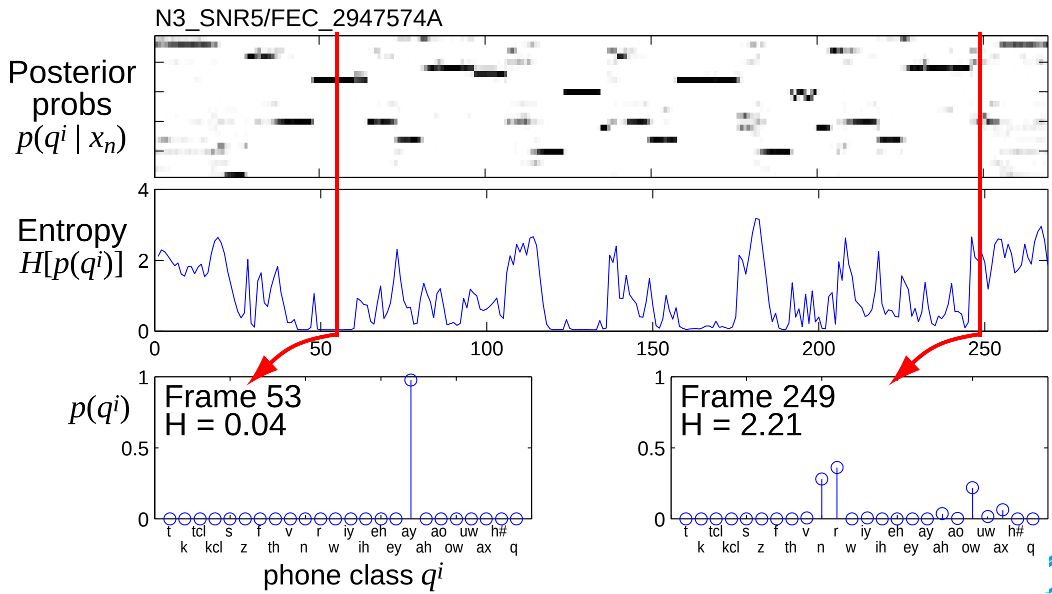
  - word graph/lattice (but: LM)




post process with different constraints

# Confidence measures

- **Can we tell which words *might* be wrong?**
- **Use** $p(X, Q|M_j) = \prod_n p(x_n|q_n)p(q_n|q_{n-1})$
  - good for comparing $M_j$s, not observations $X$
- **Discriminative measures are better**
  - e.g. *entropy* $H[p(q^i)] = -\sum_i p(q^i)\log p(q^i)$



# State of the art recognition

- **e.g. 1999 NIST ‘Broadcast News’ recognition**
  - real TV and radio news broadcasts
  - 100,000 word vocabulary
  - 150 hours of transcribed training data
- **Features of best systems (LIMSI, Cambridge)**
  - context-dependent GMM acoustic models: 3,500 states; 300,000 Gaussians
  - speaker adaptation in training and test
  - segmentation and clustering...
  - discriminative training?
- **WER Performance:**
  - overall: ~14% (300x real time)
  - studio speech: 8%
  - ‘fast’ system ~17% (10x real time)

# 2 Spoken Document Retrieval (SDR)

- **20% WER is horrible for transcription**
  - is it good for anything else?

- **Information Retrieval (IR)**
  - TREC/MUC 'spoken documents'
  - tolerant of word error rate, e.g.:

F0: THE VERY EARLY RETURNS OF THE NICARAGUAN PRESIDENTIAL ELECTION SEEMED TO FADE BEFORE THE LOCAL MAYOR ON A LOT OF LAW

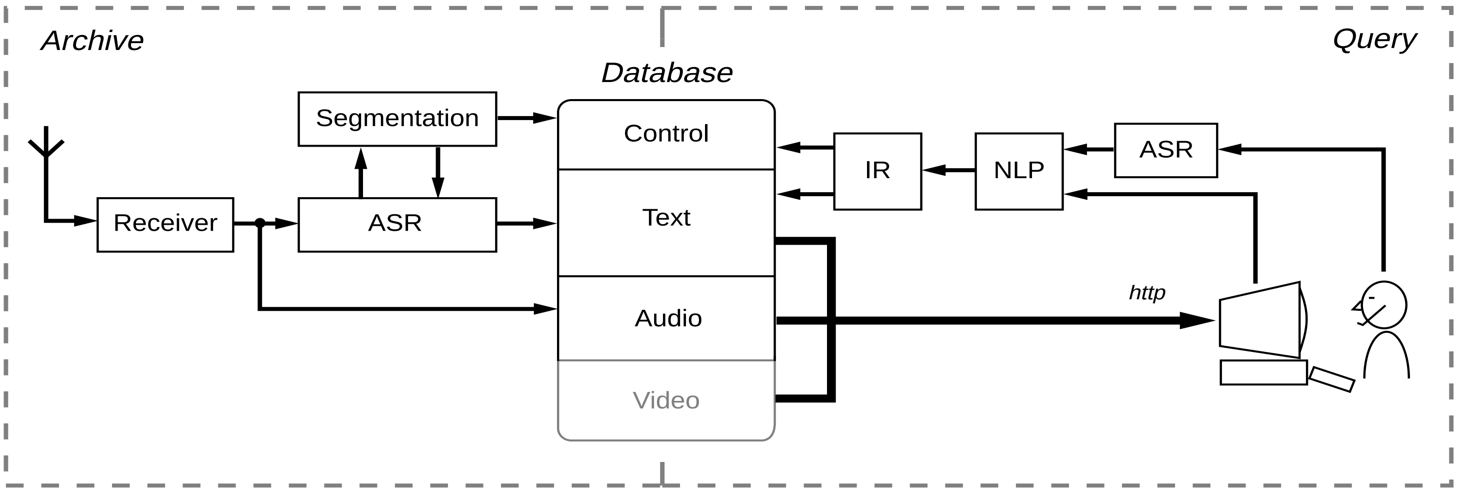
F4: AT THIS STAGE OF THE ACCOUNTING FOR SEVENTY SCOTCH ONE LEADER DANIEL ORTEGA IS IN SECOND PLACE THERE WERE TWENTY THREE PRESIDENTIAL CANDIDATES OF THE ELECTION

F5: THE LABOR MIGHT DO WELL TO REMEMBER THE LOST A MAJOR EPISODE OF TRANSATLANTIC CONNECT TO A CORPORATION IN BOTH CONSERVATIVE PARTY OFFICIALS FROM BRITAIN GOING TO WASHINGTON THEY WENT TO WOOD BUYS GEORGE BUSH ON HOW TO WIN A SECOND TO NONE IN LONDON THIS IS STEPHEN BEARD FOR MARKETPLACE

- **Promising application area**
  - document retrieval already hit-and-miss
  - plenty of untranscribed material

# The THISL SDR system

- **Original task: BBC newsroom support**



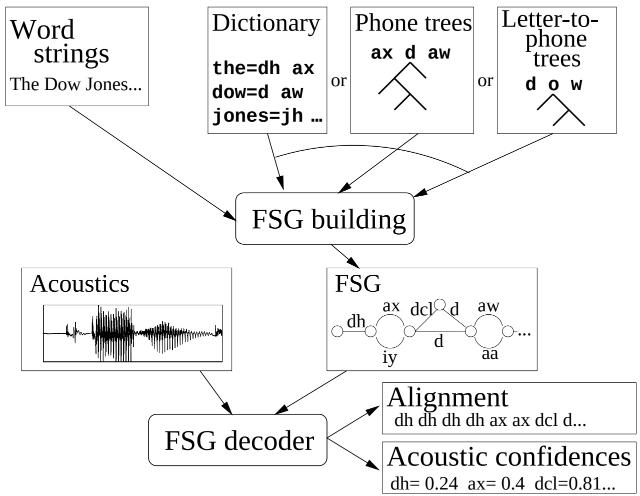

- **How to build the database:**
  - automatically record news programs ‘off air’
  - several hours per day $\rightarrow$ > 3,000 hrs
  - run recognition the whole time
  - problems storing audio!

# Building a new recognizer

- **No recognizer available for BBC English**
  - need to develop a new one based on US English Broadcast News, read British English...
- **Training set: Manual transcription of 40 hours of news**
  - word-level transcription takes > 10x real-time
  - Viterbi training, starting from read speech model
- **Language model: 200M words of US & UK newspaper archives**
- **Dictionary: Standard UK-English + extensions**
  - many novel & foreign words

# Vocabulary extension

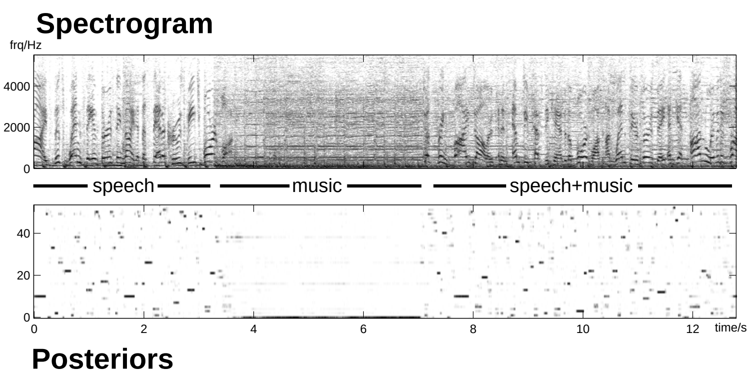
- **News always has extra words**
- **Starting point: Text-to-speech rules**
  - speech synthesizers' rules for unknown words
  - but novel words are often foreign names
- **Sources to identify new words**
  - BBC 'house style' information
- **Choose model by single acoustic example**

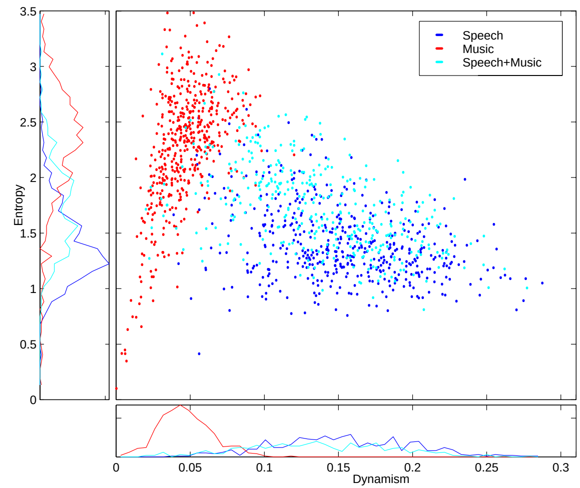



  - grab from TV subtitles?

# Audio segmentation

- **Broadcast audio includes music, noise etc.**
- **Segmentation is important for recognition**
  - speaker identity tagging, model adaptation
  - excluding nonspeech segments
- **Can use generic models of similarity/difference**
- **Look at statistics of speech model output**



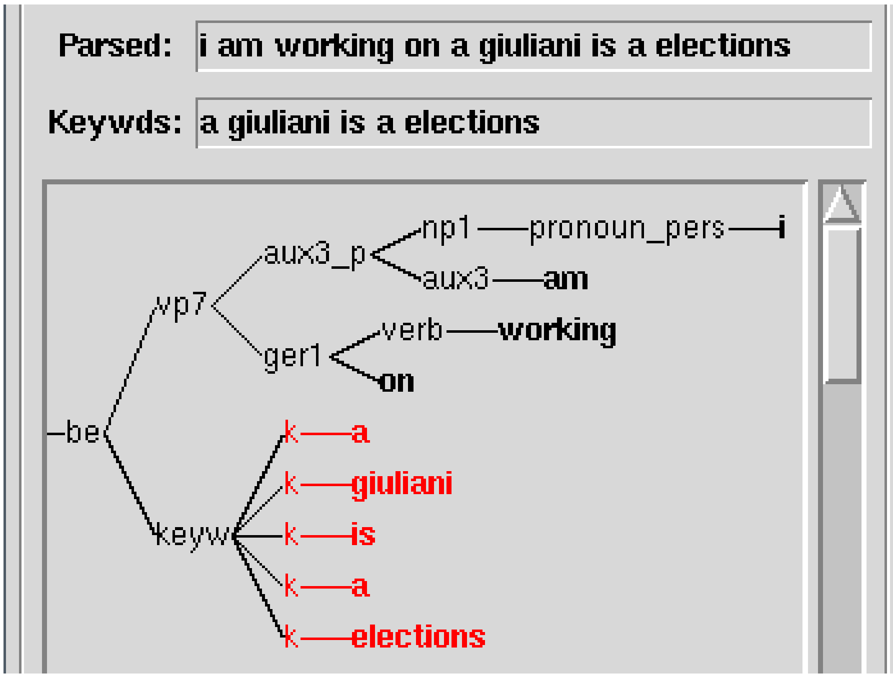

- e.g. dynamism $\frac{1}{N}\sum_n \sum_i [p(q_n^i) - p(q_{n-1}^i)]^2$

# Information retrieval: Text document IR

- **Given query terms $T_q$, document terms $T_{D(i)}$ how to *find* & *rank* documents?**
- **Standard IR uses 'inverted index' to *find*:**
  - one entry per term $T_D$, listing all documents $D(i)$ containing that term
- **Documents are *ranked* using "tf • idf"**
  - **tf** (term frequency) = how often term is in doc
  - **idf** (inverse document frequency) = how many (how few) docs contain term
- **Performance measures**
  - *precision*: (correct found)/(all found)
  - *recall*: (correct found)/(all correct)
  - mean reciprocal rank - for specific targets

# Queries in Thisl

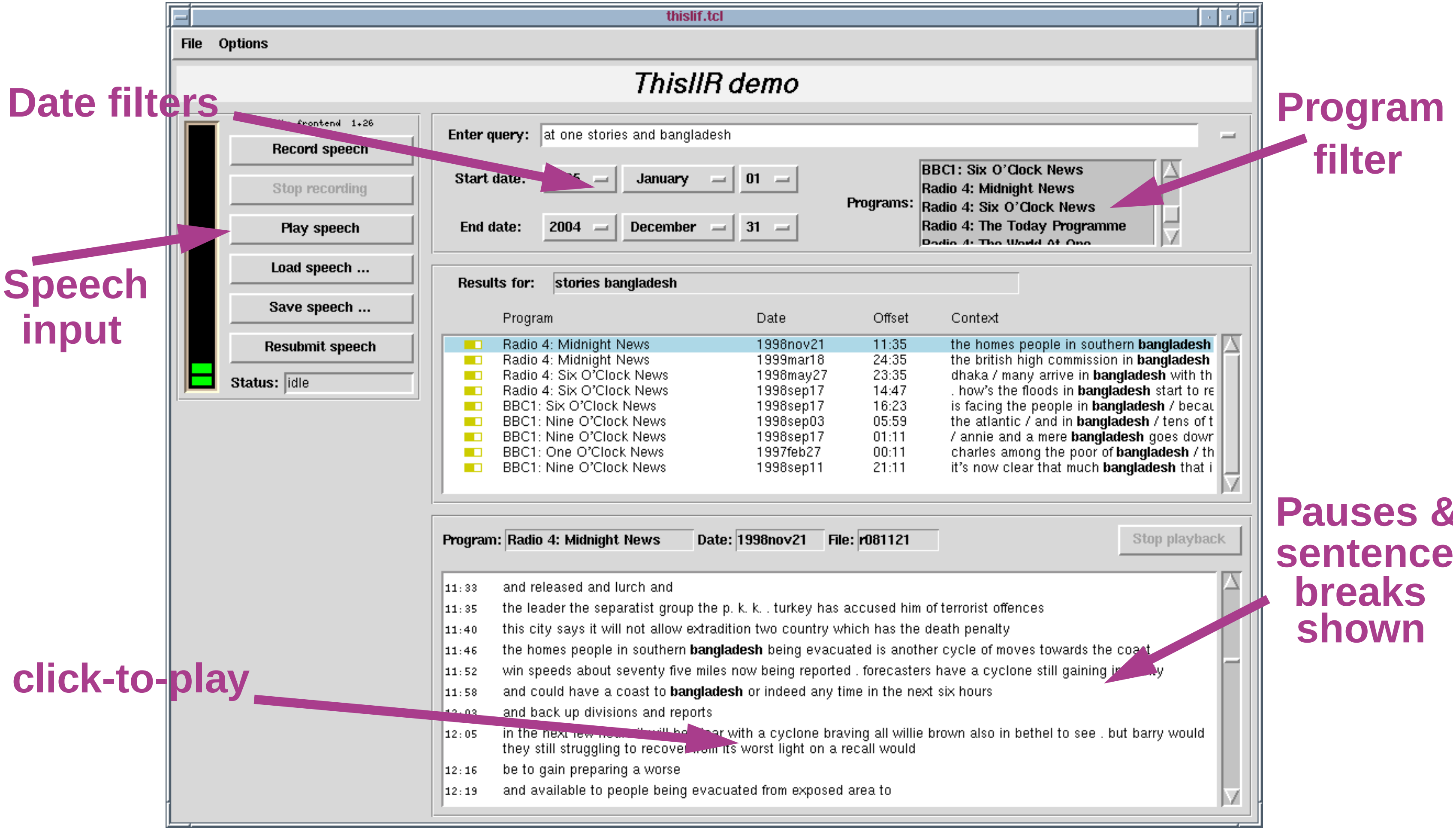
- **Original idea: speech in, speech out**
- **Try to 'understand' queries**
  - hand-built grammar:




  - .. but keywords better

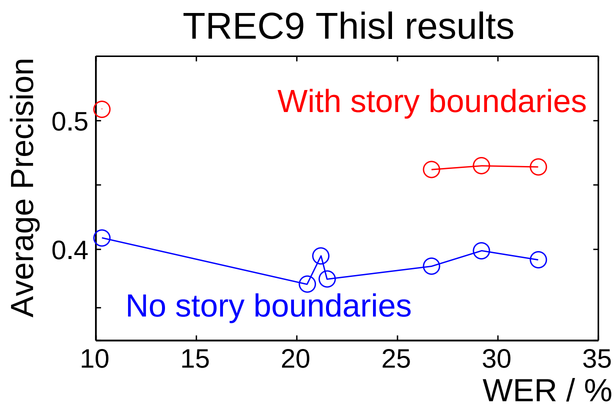
- **Phonetic matching with speech input**
  - search 'phone lattice' recognizer output?

# Thisl User Interface



# Thisl SDR performance

- **NIST Text Retrieval Conference (TREC), Spoken Documents track**
  - 500 hours of data → need fast recognition
  - set of ‘evaluation queries’ + relevance judgments
- **Components tried in different combinations**
  - different speech transcripts (subtitles, ASR)
  - different IR engines & query processing
- **Performance of systems**
  - ASR less important than IR (query expansion...)



# Outline

1. ASR wrap-up

2. Spoken Document Retrieval

3. **General audio databases**
   - Nonspeech audio retrieval
   - Audio mixtures & CASA
   - Model learning

4. Open issues

3

# Real-world audio

- **Speech is only part of the audio world**
  - word transcripts are not the whole story
- **Large audio datasets**
  - movie & TV soundtracks
  - events such as sports, news 'actualities'
  - situation-based audio 'awareness'
  - personal audio recording
- **Information from sound**
  - speaker identity, mood, interactions
  - 'events': explosions, car tires, bounces...
  - ambience: party, subway, woods
- **Applications**
  - indexing, retrieval
  - description/summarization
  - intelligent reaction

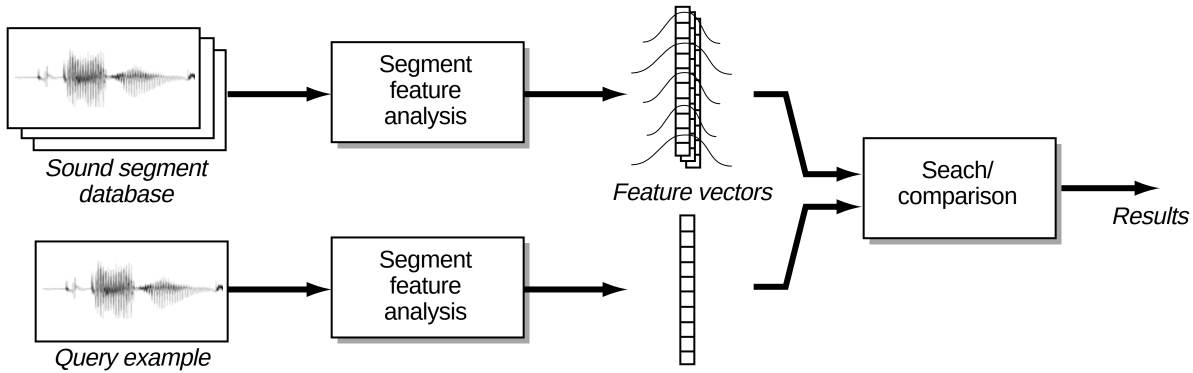
# Multimedia Description: MPEG-7

- **MPEG has produced standards for audio / video *data compression***
- **MPEG-7 is a standard for *metadata*: describing multimedia content**
  - because search & retrieval is so important
- **Defines descriptions of time-specific tags, ways to define categories, specific category instances**
- **Also defines preliminary features e.g. for audio:**
  - spectrum: centroid, spread, flatness
  - harmonicity: degree, stability
  - pitch, attack time, melody structure ...

http://www.darmstadt.gmd.de/mobile/MPEG7/Documents.html

# Muscle Fish "SoundFisher"

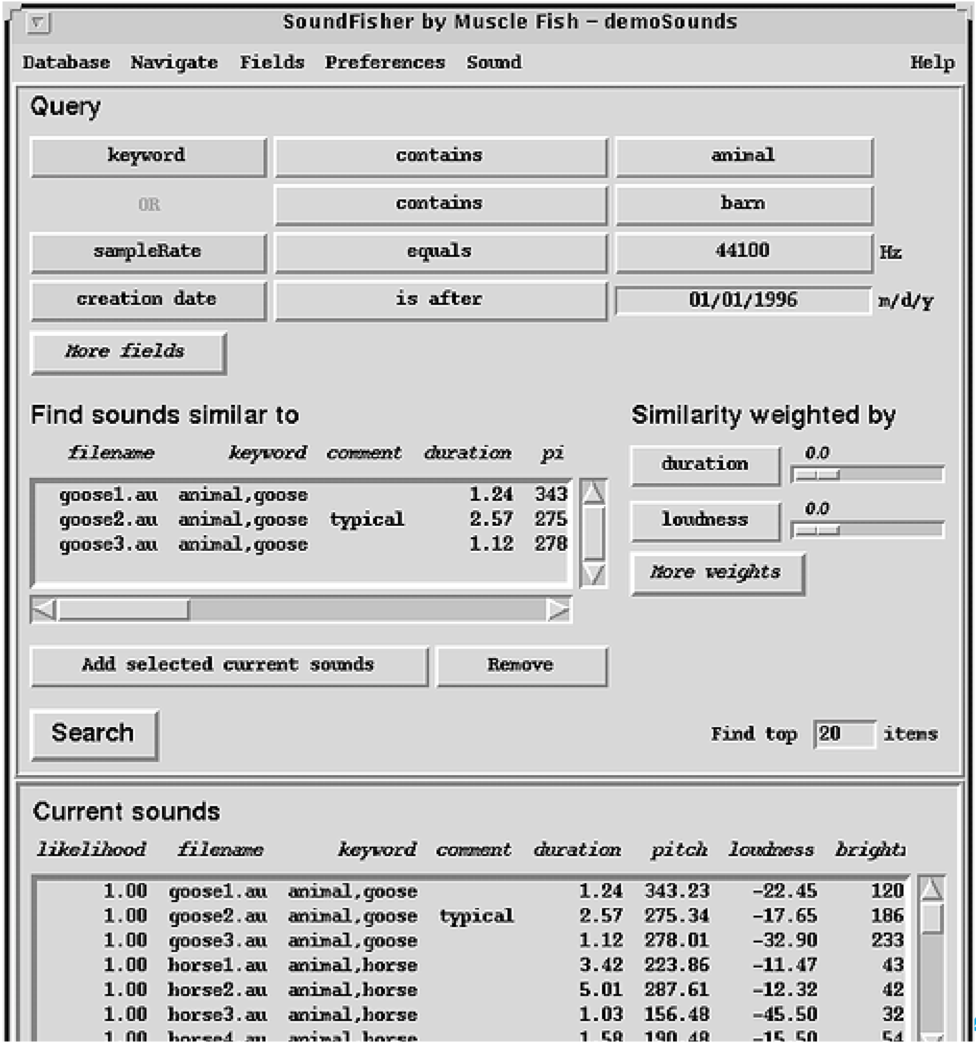
- **Access to sound effects databases**




- **Features (time series contours):**
  - loudness, brightness, pitch, cepstra
- **Query-by-example**
  - direct correlation of contours (normalized/not)
  - comparison of value histograms (time-collapsed)
- **Always *global* features**
  - a mixture of two sounds looks like neither

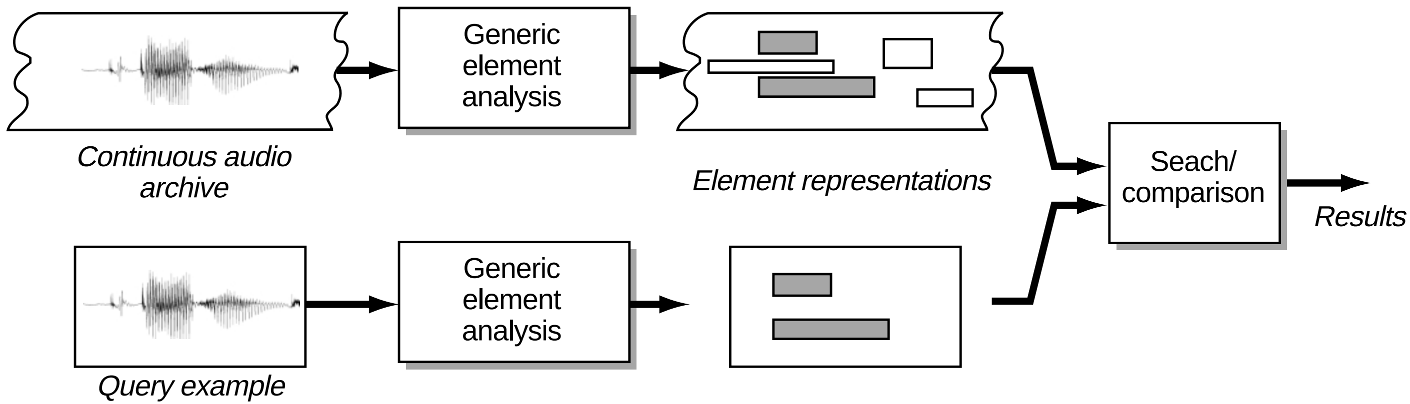
# SoundFisher user interface

- **Principle query mechanism is “sounds like”**



# Indexing for soundtracks

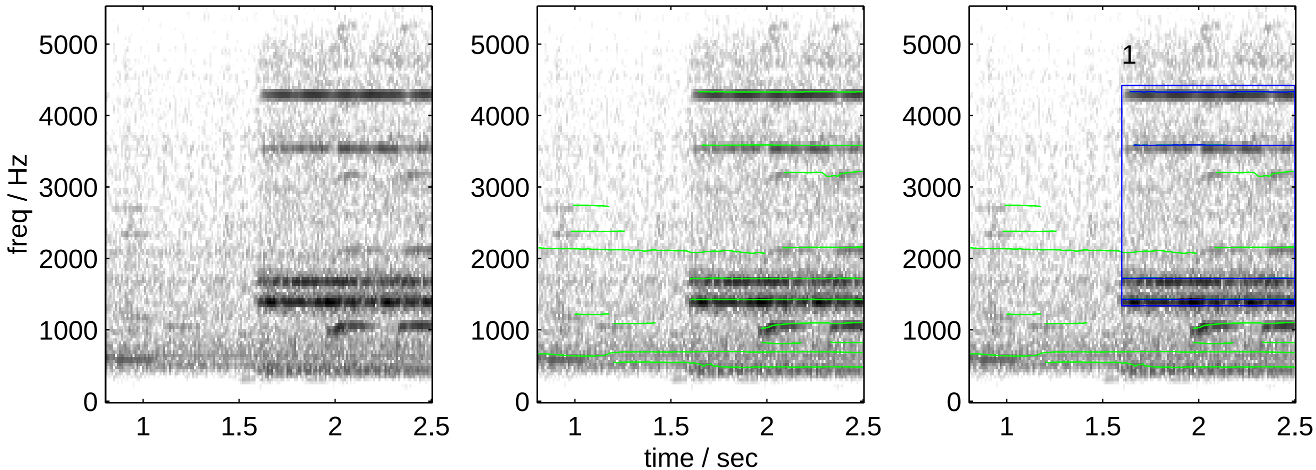
- **Any real-world audio will have multiple simultaneous sound sources**
- **Queries typically relate to one source only**
  - not a source in a particular context
- **Need to index accordingly:**




  - analyze sound into source-related elements
  - perform search & match in that domain

# Nonspeech event detection: Alarm sounds

- **Finding alarms in sound mixtures**



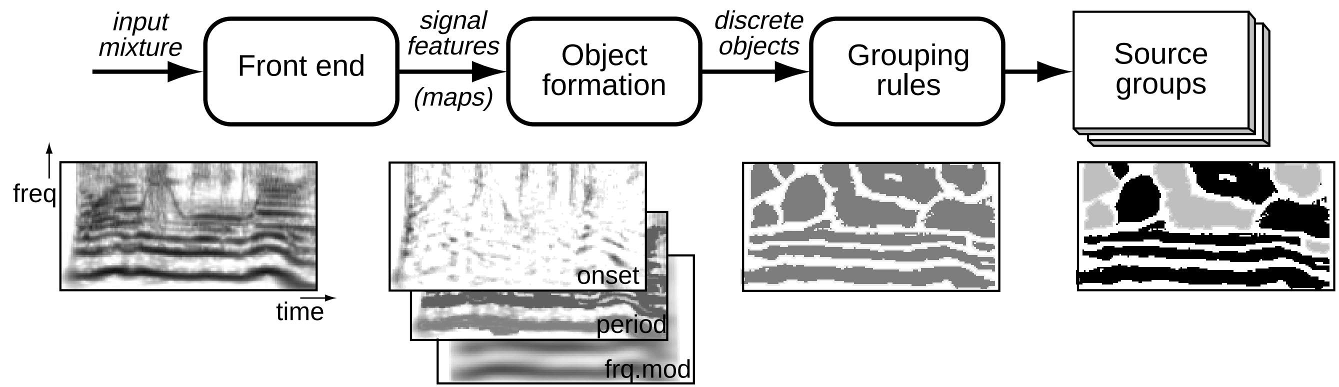

  - representation of energy in time-frequency
  - formation of atomic elements
  - grouping by common properties (onset &c.)
  - classification by attributes

$\rightarrow$ **Computational auditory scene analysis**

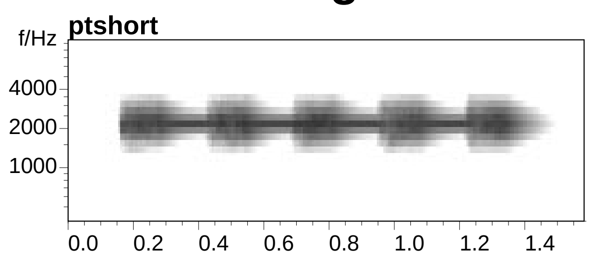
  - organizing sound mixtures by source

# Computational Auditory Scene Analysis

- **Real-world sounds come from multiple sources**
- **Psychoacoustics gives 'scene analysis' cues:**
  - common onset
  - common harmonicity
- **Suitable for computational models:**

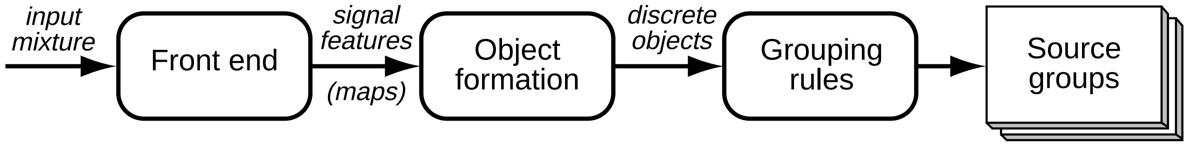



- **What about masking & 'illusions'?**

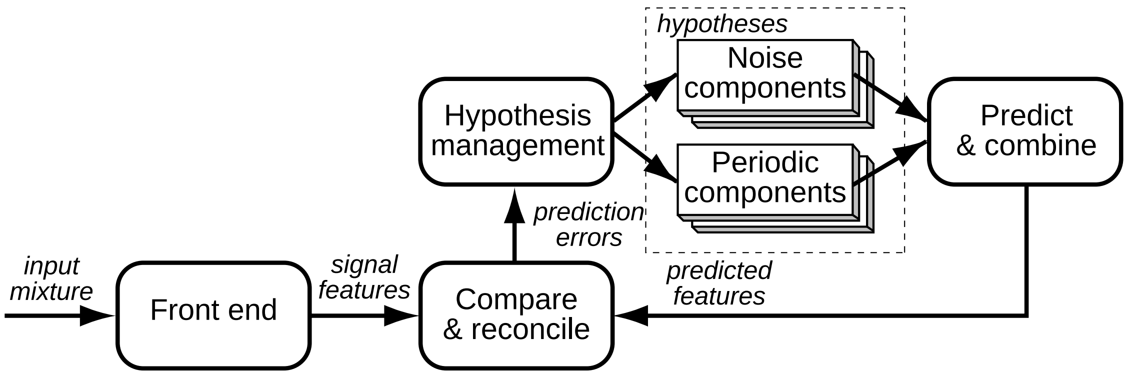


# Top-down CASA

- **Data-driven (bottom-up) fails for noisy, ambiguous sounds (most mixtures!)**

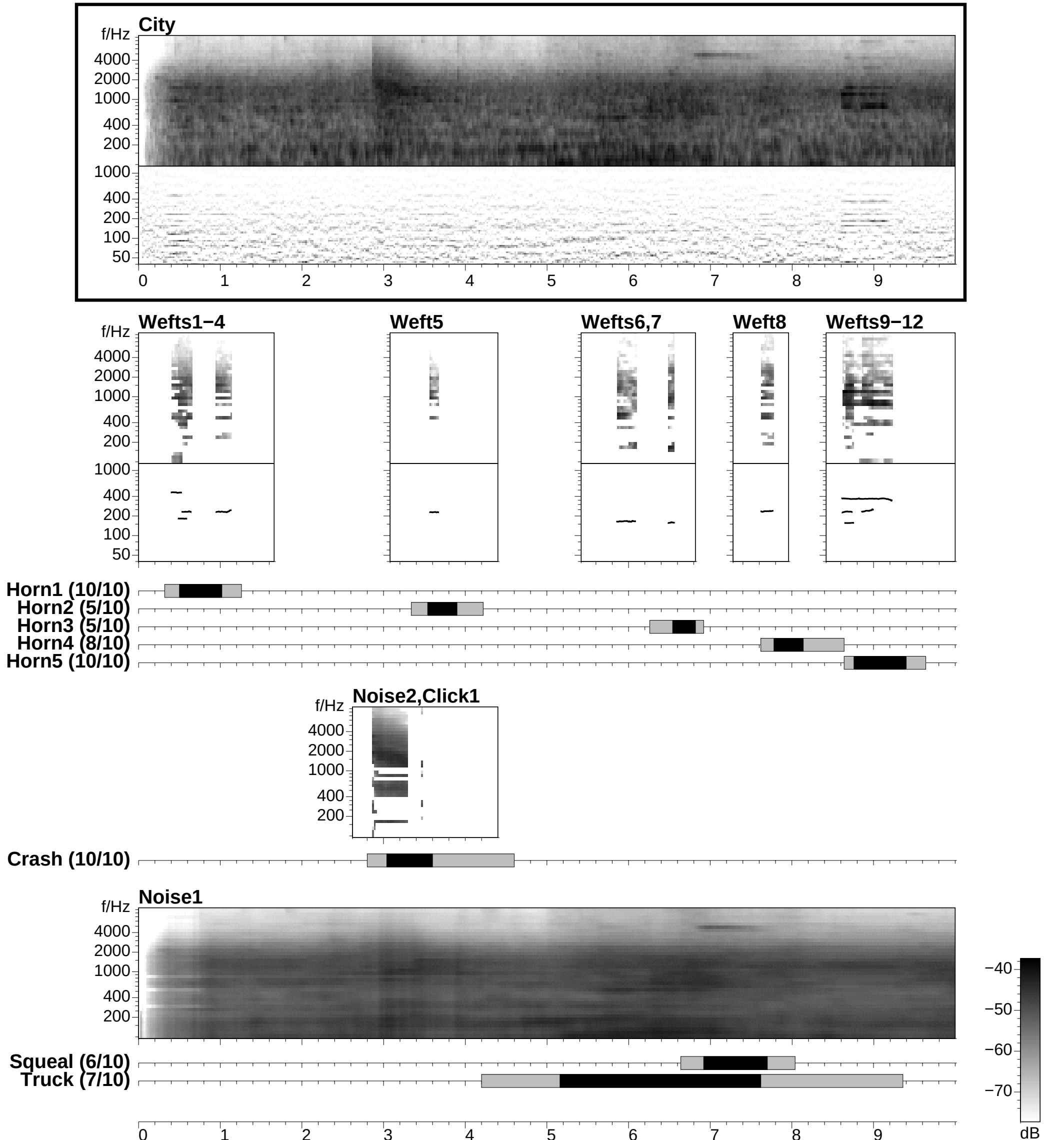



- **Need top-down constraints:**




  - fit vocabulary of generic elements to sound ... bottom of a hierarchy?
  - account for entire scene
  - driven by prediction failures
  - pursue alternative hypotheses

# CASA example



# Outline

1. ASR wrap-up
2. Spoken document retrieval
2. General audio databases
4. **Open issues**
   - Speech recognition
   - Sound source separation
   - Information extraction & visualization
   - Learning from audio

4

# Open issues 1: Speech recognition

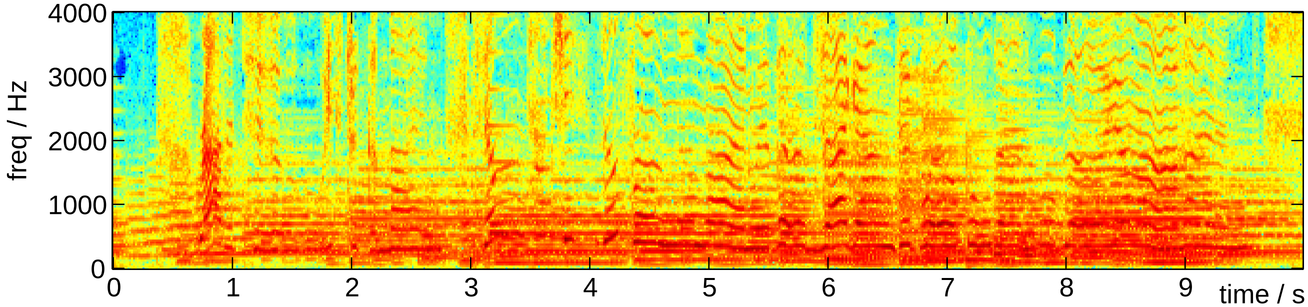
- **Speech recognition is good & improving**
  - but rate of improvement has slowed:
    BN WER: 1997=22%  1998=16%  1999=14%
  - limits to benefits of additional data?
- **Problem areas:**
  - noisy speech (meetings, cellphones)
  - informal speech (casual conversations)
  - speaker variations (style, accent)
- **Is the current approach correct?**
  - MFCC-GMM-HMM systems are optimized
  - new approaches can't compete
  - but: independence, classifier, HMMs...

# Sound mixtures

- **Real-world sound always consists of mixtures**
  - subjective experience terms of separate sources
  - ‘intelligent’ systems must do the same
- **How to separate sound sources?**
  - exact decomposition (‘blind source separation’)
  - extract cues
  - overlap, masking
    $\rightarrow$ top-down approaches, analysis-by-synthesis
- **How to represent & recognize sources?**
  - which features, attributes?
  - hierarchy of general-to-specific classes...

# Information & visualization

- **Spectrograms are OK for speech, often unsatisfactory for more complex sounds**




  - frequency axis, intensity axis – time axis?
  - separate spatial, pitch, source dimensions

- **Visualization may not be possible .. but helps us think about sound features**

- **Different representations for different aspects**
  - best for speech, music, environmental, ...

# Learning from audio

- **HMMs (EM, Baum-Welch etc.) have had a huge impact on speech, handwriting ...**
  - very good for optimizing models
  - little help for determining model structure
- **Applicable to other audio tasks?**
  - e.g. textures, ambience, vehicles, instruments
- **Problems:**
  - finding the right model structures
  - constraining what the models learn:
    initial clustering, target labelling
- **How to leverage large databases, bulk audio**
  - unsupervised acquisition of classes, features
  - the analog of infant development

# Outline

1. ASR details
2. ASR in practice
2. Real-world audio
4. Open issues

# Course retrospective

| | | | | |
|---|---|---|---|---|
| *Tools* | DSP + | Acoustics + | Perception + | Pattern recognition |
| *Signals* | Music | Speech | Other... | |
| *Applications* | Models | Coding | **Recognition** | |
| | | | | Indexing |

# Summary

- **Automatic Speech Recognizers**
  - generate wordstring hypotheses
  - can squeeze out some other information
- **Large Vocabulary recognition**
  - dictation is a limited domain
  - noisy recognition useful for indexing
  + speech understanding?
- **Recognizing nonspeech audio**
  - lots of other kinds of acoustic events
  - speech-style recognition can be applied
- **Open questions**
  - lots of things that we *know* that we *don't* know