# ICASSP 2008 survey

Presenter: Suhan Yu

# Spoken Language Understanding

- Using corpus and knowledge-based similarity measure in maximum marginal relevance for meeting summarization. *Shasha Xie, Yang Liu*.
- Extension of HVS semantic parser by allowing left-right branching. *Filip Jurcicek, Jan Svec, and Ludek Muller.*
- Acoustic classification of question turns in spontaneous speech using lexical and prosodic evidence. Sankaranarayanan Ananthakrishnan et.al.

# Using corpus and knowledge-based similarity measure in maximum marginal relevance for meeting summarization

- This paper evaluate different similarity measures in the MMR framework for meeting summarization on the ICSI meeting corpus.
  - Cosine similarity
  - Centroid score
  - Corpus-based semantic similarity
- We introduce a corpus-based measure to capture the similarity at the semantic level, and compare this method with cosine similarity and centroid score that only considers the salient words in the segments.
- The experimental results evaluated by the ROUGE.

# Maximum Marginal Relevance (MMR)

- MMR:

$$MMR\ (S_i) = \lambda \times Sim_1(S_i, D) - (1 - \lambda) \times Sim_2(S_i, Summ\ )$$

sentence

adjust the combined score

Document vector

the sentences that have been extracted into the summary

# Similarity methods

- Cosine similarity

$$sim\,(D_1, D_2) = \frac{\sum_i t_{1i} t_{2i}}{\sqrt{\sum_i t_{1i}^2} \times \sqrt{\sum_i t_{2i}^2}}$$

- Centroid Score

$$Score_{centroid}(i) = \sum_{w_i \in S_i} bool(w_j \in T) \times bool(tw(w_j) > v) \times tw(w_j)$$

- The cosine and centroid scores between a sentence and a document are all based on simple lexical matching, that is, only the words that occur in both contribute to the similarity.
- Such literal comparison can not always capture the semantic similarity of text.

# Similarity methods

- Corpus-based Semantic Similarity
  - compute the similarity score between two text segments.

$$sim(T_1, T_2) = \frac{1}{2}\left(\frac{\sum_{w \in \{T_1\}} (\max\ Sim(w, T_2) * idf(w))}{\sum_{w \in \{T_1\}} idf(w)} + \frac{\sum_{w \in \{T_2\}} (\max\ Sim(w, T_1) * idf(w))}{\sum_{w \in \{T_2\}} idf(w)}\right)$$

$$\max\ Sim(w, T_i) = \max_{w_i \in \{T_i\}} \{sim(w, w_i)\}$$

$$PMI\ (w_1, w_2) = \log_2 \frac{c(w_1\ near\ w_2)}{c(w_1) * c(w_2)}$$

In [8], Murray and Renals compared different term weighting approaches to rank the importance of the sentences (simply based on the sum of all the term weights in a sentence) for meeting summarization, and showed that TF-IDF weighting is competitive. Therefore in this study, we will use TF-IDF for term weighting and focus on the problem of how to calculate the similarity between two documents in the MMR framework.

In our experiments, we also found that different normalization methods for the cosine similarity have a great effect on the system performance. The method we adopt in this paper is to first calculate the dot product score (i.e., without the denominator in Eq 2) for $Sim_1$, then scaling it to [0,1] based on the maximum scores among all the sentences. We use the original cosine score for $Sim_2$.

$maxSim(w,T)$ is 1 if $w$ appears in $T$

# Similarity methods and its approximation

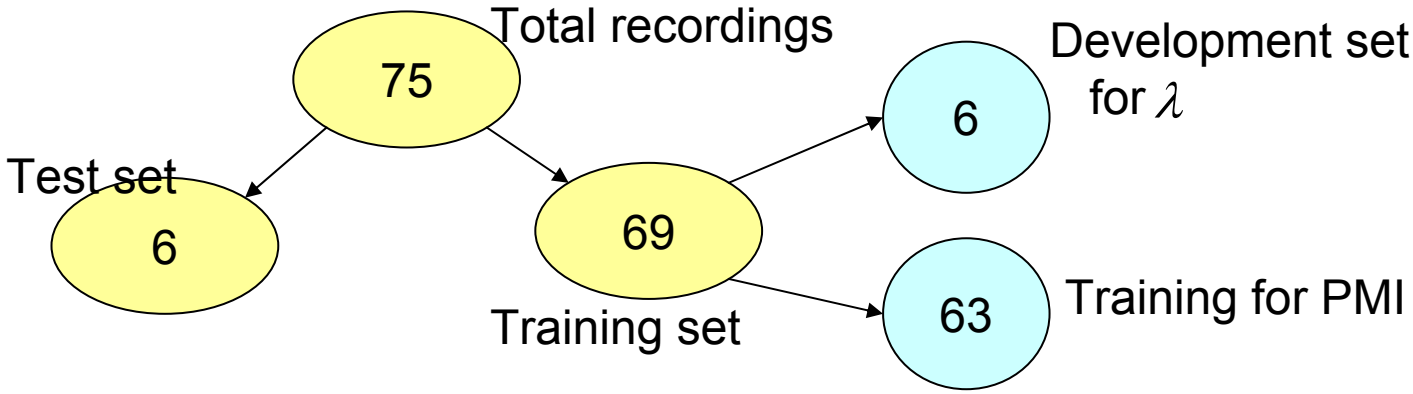
- Consider part-of-speech (POS) information.

$$\max Sim(w, T_i) = \max_{\substack{w_i \in \{T_i\} \\ pos(w_i)=pos(w)}} \{sim(w, w_i)\}$$

- Approximation in MMR computation
  - The speed of the system is especially a problem for the corpus-based similarity.
  - It is more complex and time-consuming than cosine similarity since we need to compare every word pair in the two text segments.
  - For each sentence, we calculate its similarity to all the other sentences that have a higher similarity score to the document.
  - Not to consider all the sentences in the document, but rather only a small percent of sentences (based on a predefined percentage) that have a high similarity score to the entire document.

# Data and experimental setup

- Corpus
  - ICSI meeting corpus
    - 75 recordings from natural meetings, each meeting is about an hour long.
    - These meetings have been transcribed and annotated with topic information and extractive summaries
    - The ASR output is obtained from a state-of-the-art SRI conversational telephone speech (CTS) system
    - The word error rate on the entire corpus is about 38.2%.



# Data and experimental setup

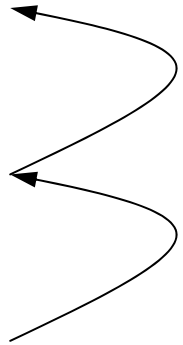
- POS tagger
  - TnT (*Trigrams'n'Tags* ) POS
  - A very efficient statistical part-of-speech tagger that is trainable on different languages and virtually any tagset.
  - http://www.coli.uni-saarland.de/~thorsten/tnt/
- Train IDF
  - IDF value are obtained from the 69 training meetings.
  - Split each of the 69 training meetings into multiple topics, and then use these new “documents” to calculate the IDF values.
  - This generates more robust estimation for IDF.

# Evaluation Measurement and Result

- Evaluation Measurement
  - ROUGE
- Experimental Result
  - Using human transcripts
  - On development data

ROUGE unigram

| $Sim_1$ | $Sim_2$ | approx_1 | approx_2 | R-1 |
|---|---|---|---|---|
| cosine | cosine | no | no | 0.60465 |
| cosine | cosine | yes | 2*perc | 0.65255 |
| centroid | cosine | no | no | 0.68011 |
| centroid | cosine | yes | no | 0.68104 |
| centroid | cosine | yes | 2*perc | 0.68274 |
| corpus | corpus | yes | 2*perc | 0.68910 |
| corpus | corpus | yes | 3*perc | 0.68443 |
| corpus_pos | corpus_pos | yes | 2*perc | 0.69316 |



# Experimental Result

- Experimental Result
  - Using human transcripts
  - On test data

| $Sim_1$ | $Sim_2$ | approx_1 | approx_2 | R-1 |
|---|---|---|---|---|
| cosine | cosine | no | no | 0.58843 |
| cosine | cosine | yes | 2*perc | 0.65300 |
| centroid | cosine | no | no | 0.68938 |
| centroid | cosine | yes | no | 0.68688 |
| centroid | cosine | yes | 2*perc | 0.69103 |
| corpus | corpus | yes | 2*perc | 0.69274 |
| corpus_pos | corpus_pos | yes | 2*perc | 0.71243 |

# Experimental Result

- Experimental Result
  - On ASR output

| $Sim_1$ | $Sim_2$ | approx_1 | approx_2 | R-1 |
|---|---|---|---|---|
| cosine | cosine | no | no | 0.51425 |
| cosine | cosine | yes | 2*perc | 0.60621 |
| centroid | cosine | yes | 2*perc | 0.65024 |
| corpus | corpus | yes | 2*perc | 0.65129 |
| corpus_pos | corpus_pos | yes | 2*perc | 0.61733 |

  - the POS tagging accuracy for the ASR transcripts is relatively low.

# Conclusion

- This paper have evaluated different similarity measures under the MMR framework for meeting summarization.
  - The centroid score focuses on the salient words of a text segment, ignoring words with lower TF-IDF values. (using threshold)
  - The corpus-based semantic approach estimates the similarity of two segments based on their word distribution on a large corpus.
- These methods outperform the commonly used cosine similarity both on manual and ASR transcripts.
- Using approximation in MMR does not hurt performance, while significantly increasing the speed.
- Future work
  - evaluate the effect from automatic sentence segmentation
  - Meeting recordings contain rich information such as multiple speakers and prosody.

# Extension of HVS semantic parser by allowing left-right branching

- This paper focus on the statistical semantic parsing.
- A semantic concept is considered to be a basic unit of a particular meaning.

$$S^* = \arg\max_S P(S \mid W) = \arg\max_S P(W \mid S)P(S)$$

Observation sequence

$$W = w_1, w_2, \ldots, w_T$$

Sequence of concept

$$S = c_1, c_2, \ldots, c_T$$

Lexical model

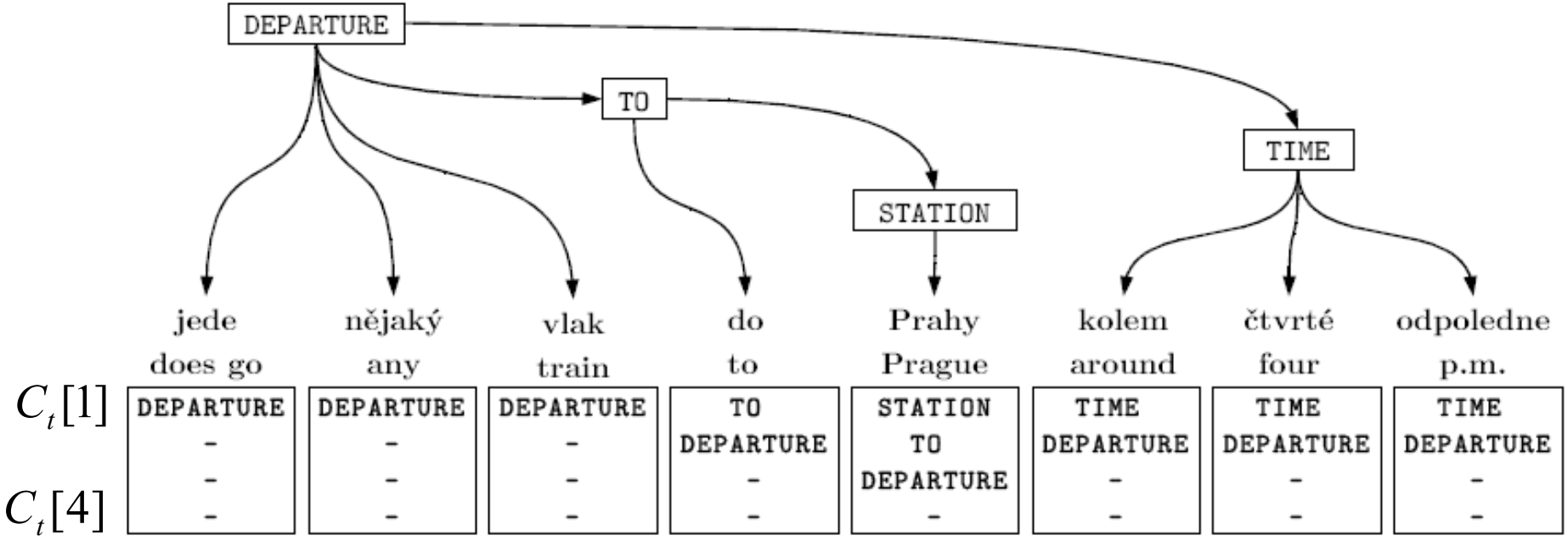
semantic model

# Extension of HVS semantic parser by allowing left-right branching

- HVS parser
  - 2005 proposed by He and Young.
  - Hidden vector state parser.
  - allows to generate right-branching semantic trees.
- This paper proposed an extension of the HVS parser
  - generate not only right-branching semantic trees but also limited left-branching semantic trees.
  - Idea comes from different language with different properties.
    - Right branching language
      - Spanish: adjectives usually follow nouns, direct objects follow verbs.
    - Left branching language
      - Japanese: adjectives precede nouns, direct objects come before verbs.
    - English shows left branching at the level of noun phrases but it is mostly right-branching at the sentence level.

# Hidden vector state parser

- The HVS parser is an approximation of a pushdown automaton. (pushdown automaton (PDA) is a finite automaton that can make use of a stack containing data.)
- Semantic tree: Departure ( To ( Station ) ,Time)




$t$ : *position*

1,2,3,4 : *four concepts*

# Hidden vector state parser

- Viewing each vector state as a hidden variable, the whole parse tree can be converted into a first order vector state Markov model, this is the HVS model.

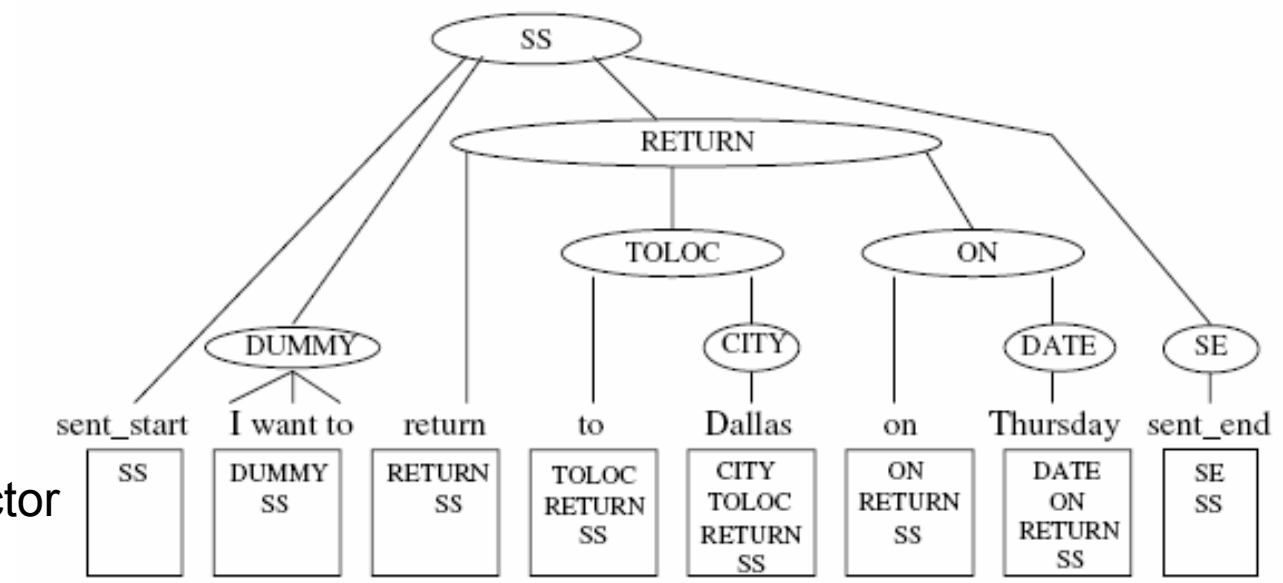



Fig. 2. Example of a parse tree and its vector state equivalent.

# Hidden vector state parser

$$S^* = \arg\max_S P(S \mid W) = \arg\max_S P(W \mid S)P(S)$$

$$P(W \mid S) = \prod_{t=1}^{T} P(w_t \mid c_t[1,...,4])$$

$$P(S) = \prod_{t=1}^{T} P(pop_t \mid c_{t-1}[1,...,4]) P(c_t[1]c_t[2,..,4])$$

represents a model for popping 0 to 4 concepts from the stack

State transition

$pop_t$ defines the number of concepts which will be popped off the stack.

# Left-Right-branching parsing

- Modification of the HVS parser
  - Parser with probabilistic pushing (HVS-PP)
  - pushing operation which takes values 0 for pushing no concept and 1 for pushing one concept onto the stack.

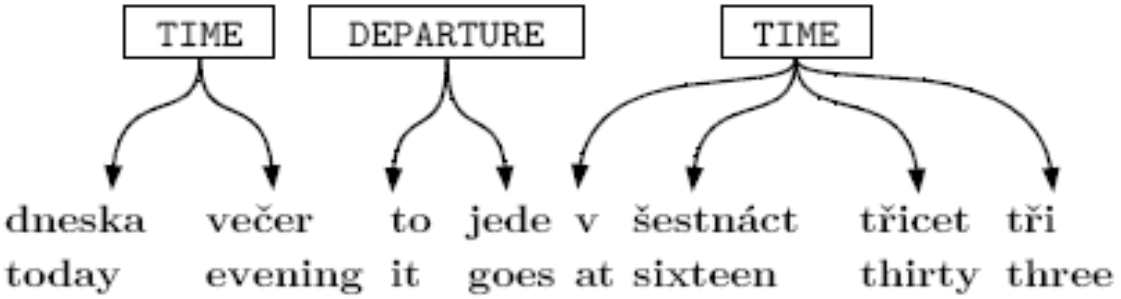
$$P(S) = \prod_{t=1}^{T} P(pop_t|c_{t-1}[1,\dots 4])P(push_t|c_{t-1}[1,\dots 4]) \cdot \begin{cases} 1 & \text{if } push_t = 0 \\ P(c_t[1]|c_t[2,\dots 4]) & \text{if } push_t = 1 \end{cases} \quad (4)$$

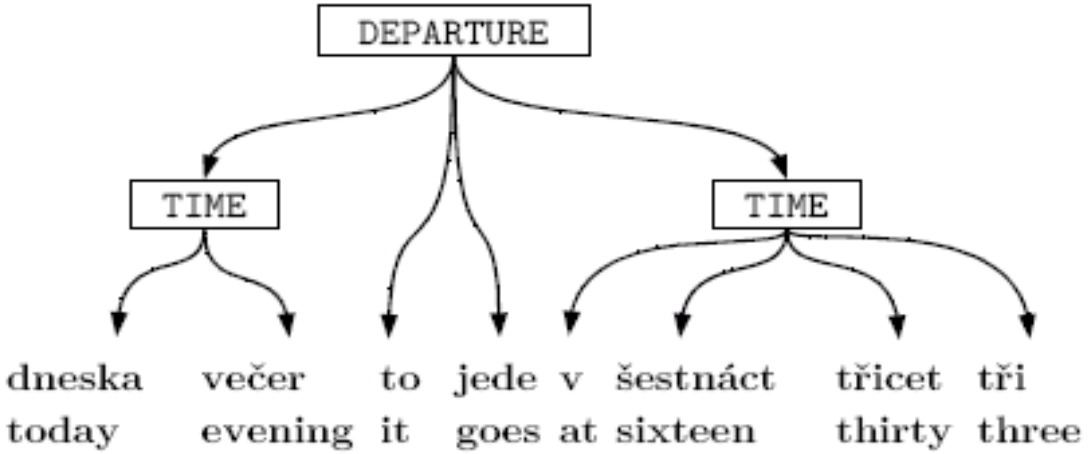
  - Left-right-branching HVS (LRB-HVS)

$$P(S) = \prod_{t=1}^{T} P(pop_t|c_{t-1}[1,\dots 4])P(push_t|c_{t-1}[1,\dots 4]) \cdot \begin{cases} 1 & \text{if } push_t = 0 \\ P(c_t[1]|c_t[2,\dots 4]) & \text{if } push_t = 1 \\ P(c_t[1]|c_t[2,\dots 4])P(c_t[2]|c_t[3,4]) & \text{if } push_t = 2 \end{cases}$$

Push not only one concept

# Left-Right-branching parsing

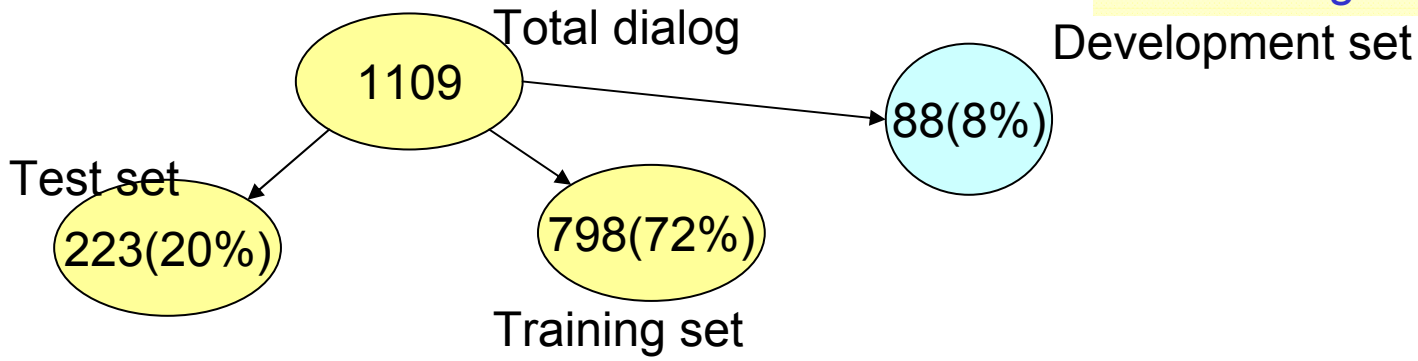


# Experiments

- Corpus
  - The semantic parsers evaluated in this article were trained and tested on the Czech human-human train timetable (HHTT) dialog corpus.
  - 1109 dialogs, 17900 utterances in total.
  - 2872 words.
  - 35 semantic concept.

The development data were used for finding the optimal concept insertion penalties and the optimal semantic model weights



# Experiments

- Semantic accuracy

The number of exactly match the reference

$$SAcc = \frac{E}{N} \cdot 100\%$$

The number of evaluated semantics

- Concept accuracy

$$CAcc = \frac{N - S - D - I}{N} \cdot 100\%$$

N is the number of concepts in the reference semantics,
S is the number of substitutions,
D is the number of deletions
I is the number of insertions.

| | Test data | | | Development data | | |
|---|---|---|---|---|---|---|
| | SAcc | CAcc | $p$-value | SAcc | CAcc | $p$-value |
| baseline | 50.4 | 64.9 | | 52.8 | 67.0 | |
| HVS-PP | 54.1 | 67.2 | $< 0.01$ | 56.6 | 68.4 | $< 0.01$ |
| LRB-HVS | 58.3 | 69.3 | $< 0.01$ | 60.1 | 70.6 | $< 0.01$ |

# Automatic classification of question turns in spontaneous speech using lexical and prosodic evidence

- Spontaneous interaction between humans is characterized by various types of speech acts, including but not limited to questions, statements and exclamatory phrases.
- This paper focus on a more universal subset of the speech act categorization problem that of distinguishing question-bearing turns from other types of utterances in spontaneous speech.
- This paper present a system that uses prosodic and lexical evidence to detect question turns in multi-party spontaneous speech using two different techniques:

# Acoustic-prosodic classifier

- Acoustic features
  - F0 values
  - short-time energy
  - zero-crossing rate (ZCR)
  - computed every 10ms
  - extracted a total of 12 prosodic features based on the above parameters.
  - Using Weka toolkit to rank the features in order of their importance for classification.
  - F0 range within the terminal window is the most informative feature

**Table 1.** Acoustic-prosodic features in order of decreasing information gain for question turn classification

| Feature | Description |
|---|---|
| rng_val | F0 range |
| min_val | minimum F0 |
| avg_val | average F0 |
| max_val | maximum F0 |
| $a_1$ | F0 slope |
| zcr_$a_2$ | 2nd order term of ZCR polynomial fit |
| eng_$a_1$ | slope of short-time energy |
| sd_val | F0 standard deviaton |
| perc_diff | % difference between terminal avg. F0 to overall avg. F0 |
| eng_$a_2$ | 2nd order term of short-time energy polynomial fit |
| zcr_$a_1$ | slope of ZCR |
| $a_2$ | 2nd order term of F0 polynomial fit |

# Acoustic-Prosodic classifier

- Acoustic classifiers
  - GMM
    - trained 5-mixture, diagonal covariance GMMs for question and non-question
  - Multilayer perceptron classifier
    - trained with 20 hidden nodes and 2 output nodes with softmax activation that provided class posterior probabilities.

# Lexical classifiers

- Although F0-related prosodic features are useful for question turn classification, many types of questions do not exhibit a rising intonation.
  - *why, who, which*, etc. are usually characterized by a falling F0 contour.
- Language model classifier
  - capturing words and phrases that are commonly found in questions.
  - trigram LMs
    - one for each class, from the training data using the SRILM toolkit.
    - For each test utterance, we computed the log probability of the text given the two LMs.

# Lexical classifiers

- Bag-of-words classifier
  - CMU BOW toolkit
  - each utterance is described by a feature vector that contains counts of each vocabulary item that occurs in it.

**Table 2.** Discriminating words

| 1-grams | 1+2-grams |
|---|---|
| yeah | what |
| what | yeah |
| you | you |
| mmhmm | do;you |
| do | do |
| how | how |
| is | mmhmm |
| or | are;we |
| the | is;it |
| are | is |

# Experimental Results

- Corpus
  - ICSI meeting corpus
  - 75 meetings
  - total of 22,511 turns, of which 2,223 were question bearing turns and the remaining 20,288 were non-questions.

**Table 3**. Question classification performance

| Method | Accuracy |
|---|---|
| Chance | 50.0% |
| Acoustic (GMM) | 55.4% |
| Acoustic (MLP) | 61.0% |
| Lexical (LM) | 69.9% |
| MLP + LM | 71.2% |
| Lexical (BOW) | 71.3% |
| BOW + Acoustic | **71.9%** |

500 samples each
of question and non-question

# Experimental Results

- the effect of errors in the text transcription on classification performance.