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.*
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:

$$MMR (S_i) = \lambda \times \text{Sim}_1 (S_i, D) - (1 - \lambda) \times \text{Sim}_2 (S_i, \text{Summ})$$

- Adjust the combined score

- The sentences that have been extracted into the summary

- Sentence

- Document vector
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

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

• 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.

\[
sim(T_1, T_2) = \frac{1}{2} \left( \sum_{w \in T_1} \left( \max \ Sim(w, T_2) \times \text{idf}(w) \right) \right) \]

\[
+ \frac{1}{2} \sum_{w \in T_2} \left( \max \ Sim(w, T_1) \times \text{idf}(w) \right)
\]

\[
\sum_{w \in T_1} \text{idf}(w)
\]

\[
\sum_{w \in T_2} \text{idf}(w)
\]

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

PMI (w_1, w_2) = \log_2 \frac{c(w_1 \text{ near } w_2)}{c(w_1) \times c(w_2)}

maxSim(w, T) is 1 if w appears in T
Similarity methods and its approximation

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

\[
\max Sim(w, T_i) = \max_{w_i \in \{T_i\} \atop \text{pos}(w_i) = \text{pos}(w)} \{sim(w, w_i)\}
\]

• 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%.

- Total recordings: 75
  - Test set: 6
  - Training set: 69
  - Development set: 6
  - Training for PMI: 63
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/](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

<table>
<thead>
<tr>
<th>Sim₁</th>
<th>Sim₂</th>
<th>approx.₁</th>
<th>approx.₂</th>
<th>R-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>cosine</td>
<td>cosine</td>
<td>no</td>
<td>no</td>
<td>0.60465</td>
</tr>
<tr>
<td>cosine</td>
<td>cosine</td>
<td>yes</td>
<td>2*perc</td>
<td>0.65255</td>
</tr>
<tr>
<td>centroid</td>
<td>cosine</td>
<td>no</td>
<td>no</td>
<td>0.68011</td>
</tr>
<tr>
<td>centroid</td>
<td>cosine</td>
<td>yes</td>
<td>no</td>
<td>0.68104</td>
</tr>
<tr>
<td>centroid</td>
<td>cosine</td>
<td>yes</td>
<td>2*perc</td>
<td>0.68274</td>
</tr>
<tr>
<td>corpus</td>
<td>corpus</td>
<td>yes</td>
<td>2*perc</td>
<td>0.68910</td>
</tr>
<tr>
<td>corpus</td>
<td>corpus</td>
<td>yes</td>
<td>3*perc</td>
<td>0.68443</td>
</tr>
<tr>
<td>corpus_pos</td>
<td>corpus_pos</td>
<td>yes</td>
<td>2*perc</td>
<td>0.69316</td>
</tr>
</tbody>
</table>
### Experimental Result

- Using human transcripts
- On test data

<table>
<thead>
<tr>
<th>$Sim_1$</th>
<th>$Sim_2$</th>
<th>approx.1</th>
<th>approx.2</th>
<th>R-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>cosine</td>
<td>cosine</td>
<td>no</td>
<td>no</td>
<td>0.58843</td>
</tr>
<tr>
<td>cosine</td>
<td>cosine</td>
<td>yes</td>
<td>2*perc</td>
<td>0.65300</td>
</tr>
<tr>
<td>centroid</td>
<td>cosine</td>
<td>no</td>
<td>no</td>
<td>0.68938</td>
</tr>
<tr>
<td>centroid</td>
<td>cosine</td>
<td>yes</td>
<td>no</td>
<td>0.68688</td>
</tr>
<tr>
<td>centroid</td>
<td>cosine</td>
<td>yes</td>
<td>2*perc</td>
<td>0.69103</td>
</tr>
<tr>
<td>corpus</td>
<td>corpus</td>
<td>yes</td>
<td>2*perc</td>
<td>0.69274</td>
</tr>
<tr>
<td>corpus_pos</td>
<td>corpus_pos</td>
<td>yes</td>
<td>2*perc</td>
<td>0.71243</td>
</tr>
</tbody>
</table>
Experimental Result

- On ASR output

<table>
<thead>
<tr>
<th>Sim₁</th>
<th>Sim₂</th>
<th>approx.₁</th>
<th>approx.₂</th>
<th>R-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>cosine</td>
<td>cosine</td>
<td>no</td>
<td>no</td>
<td>0.51425</td>
</tr>
<tr>
<td>cosine</td>
<td>cosine</td>
<td>yes</td>
<td>2*perc</td>
<td>0.60621</td>
</tr>
<tr>
<td>centroid</td>
<td>cosine</td>
<td>yes</td>
<td>2*perc</td>
<td>0.65024</td>
</tr>
<tr>
<td>corpus</td>
<td>corpus</td>
<td>yes</td>
<td>2*perc</td>
<td>0.65129</td>
</tr>
<tr>
<td>corpus_pos</td>
<td>corpus_pos</td>
<td>yes</td>
<td>2*perc</td>
<td>0.61733</td>
</tr>
</tbody>
</table>

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

- This paper has 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 focuses 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
\[
S = c_1, c_2, \ldots, c_T
\]
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)

![Semantic Tree Diagram]

<table>
<thead>
<tr>
<th>t</th>
<th>position</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 2, 3, 4</td>
<td>four concepts</td>
</tr>
</tbody>
</table>
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.
Hidden vector state parser

\[ S^* = \arg \max_s P(S \mid W) = \arg \max_s P(W \mid S)P(S) \]

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

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

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.

\[
P(S) = \prod_{t=1}^{T} P(\text{pop}_t | c_{t-1}[1, \ldots 4]) P(\text{push}_t | c_{t-1}[1, \ldots 4])
\]

\[
\cdot \begin{cases} 
  1 & \text{if } \text{push}_t = 0 \\
  P(c_t[1]|c_t[2, \ldots 4]) & \text{if } \text{push}_t = 1 \\
  P(c_t[1]|c_t[2, \ldots 4]) P(c_t[2]|c_t[3, 4]) & \text{if } \text{push}_t = 2
\end{cases}
\]

– Left-right-branching HVS (LRB-HVS)

\[
P(S) = \prod_{t=1}^{T} P(\text{pop}_t | c_{t-1}[1, \ldots 4]) P(\text{push}_t | c_{t-1}[1, \ldots 4])
\]

\[
\cdot \begin{cases} 
  1 & \text{if } \text{push}_t = 0 \\
  P(c_t[1]|c_t[2, \ldots 4]) & \text{if } \text{push}_t = 1 \\
  P(c_t[1]|c_t[2, \ldots 4]) P(c_t[2]|c_t[3, 4]) & \text{if } \text{push}_t = 2
\end{cases}
\]
Left-Right-branching parsing

dneska večer to jede v šestnáct třicet tři
today evening it goes at sixteen thirty three
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.

<table>
<thead>
<tr>
<th>Test data</th>
<th>Development data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SAcc</td>
</tr>
<tr>
<td>baseline</td>
<td>50.4</td>
</tr>
<tr>
<td>HVS-PP</td>
<td>54.1</td>
</tr>
<tr>
<td>LRB-HVS</td>
<td>58.3</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>52.8</td>
</tr>
<tr>
<td></td>
<td>56.6</td>
</tr>
<tr>
<td></td>
<td>60.1</td>
</tr>
</tbody>
</table>
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 focuses 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 presents 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>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>min_val</td>
<td>minimum F0</td>
</tr>
<tr>
<td>avg_val</td>
<td>average F0</td>
</tr>
<tr>
<td>max_val</td>
<td>maximum F0</td>
</tr>
<tr>
<td>$a_1$</td>
<td>F0 slope</td>
</tr>
<tr>
<td>zcr$_{a_2}$</td>
<td>2nd order term of ZCR polynomial fit</td>
</tr>
<tr>
<td>eng$_{a_1}$</td>
<td>slope of short-time energy</td>
</tr>
<tr>
<td>sd_val</td>
<td>F0 standard deviation</td>
</tr>
<tr>
<td>perc_diff</td>
<td>% difference between terminal avg. F0 to overall avg. F0</td>
</tr>
<tr>
<td>eng$_{a_2}$</td>
<td>2nd order term of short-time energy polynomial fit</td>
</tr>
<tr>
<td>zcr$_{a_1}$</td>
<td>slope of ZCR</td>
</tr>
<tr>
<td>$a_2$</td>
<td>2nd order term of F0 polynomial fit</td>
</tr>
</tbody>
</table>
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>
<thead>
<tr>
<th>1-grams</th>
<th>1+2-grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>yeah</td>
<td>what</td>
</tr>
<tr>
<td>what</td>
<td>yeah</td>
</tr>
<tr>
<td>you</td>
<td>you</td>
</tr>
<tr>
<td>mmmHmm</td>
<td>do;you</td>
</tr>
<tr>
<td>do</td>
<td>do</td>
</tr>
<tr>
<td>how</td>
<td>how</td>
</tr>
<tr>
<td>is</td>
<td>mmmHmm</td>
</tr>
<tr>
<td>or</td>
<td>are;we</td>
</tr>
<tr>
<td>the</td>
<td>is;it</td>
</tr>
<tr>
<td>are</td>
<td>is</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chance</td>
<td>50.0%</td>
</tr>
<tr>
<td>Acoustic (GMM)</td>
<td>55.4%</td>
</tr>
<tr>
<td>Acoustic (MLP)</td>
<td>61.0%</td>
</tr>
<tr>
<td>Lexical (LM)</td>
<td>69.9%</td>
</tr>
<tr>
<td>MLP + LM</td>
<td>71.2%</td>
</tr>
<tr>
<td>Lexical (BOW)</td>
<td>71.3%</td>
</tr>
<tr>
<td>BOW + Acoustic</td>
<td>71.9%</td>
</tr>
</tbody>
</table>

500 samples each of question and non-question
Experimental Results

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