Question Detection Using Decision Tree Models

By Griffin Holt, Henry Davis, & Tysum Ruchti
Research

- Focus: Question identification through *sound* rather than *word content*
- Most Commonly Reported Success: Adaboost Random Forests & RNNs
- Used statistics of various audio features
  - Pitch (especially towards the end of a clip) = Important
- Improved results when *sound* analysis combined with *word content* analysis
The Learning Problem

1. Bucket of Models
   - Decision Trees
   - Random Forests

2. Data
   Audio files with timestamps at beginning of each question

3. Notion of “Best
   1. Gini Impurity
   2. Adaboost Error Function
   3. ROC AUC Score
1. The Bucket of Models

The Smaller Bucket: Decision Trees

1. The Bucket of Models

The Smaller Bucket: Decision Trees

Tree Visualization

- **p**etala l**eng**th (cm) <= 2.45
  - gini = 0.667
  - samples = 150
  - value = [50, 50, 50]
  - class = setosa

- **g**ini = 0.0
  - samples = 50
  - value = [50, 0, 0]
  - class = setosa

- **p**etal width (cm) <= 1.75
  - gini = 0.5
  - samples = 100
  - value = [0, 50, 50]
  - class = versicolor

- **g**ini = 0.168
  - samples = 54
  - value = [0, 49, 5]
  - class = versicolor

- **g**ini = 0.043
  - samples = 46
  - value = [0, 1, 45]
  - class = virginica

Division of Feature Space Visualization

- **D**epth = 0

- **D**epth = 1

- **D**epth = 2

---

1. The Bucket of Models

The Bigger Bucket: Random Forests of 200 Decision Trees
2. Data

28 Videos, each w/ XML files containing annotations
2. Data

- Rolling Windows of 5-second clips every 500 ms
- ~105,000 total clips created from 28 videos
- Randomly selected 20,000 clips
  - Trained on 16,000 (80%)
  - Tested on 4,000 (20%)
2. Data

Jitter

\[ \frac{N}{N-1} \frac{\sum_{i=1}^{N-1} |T_{i+1} - T_i|}{\sum_{i=1}^{N} T_i} \]

Shimmer

\[ \frac{N}{N-1} \frac{\sum_{i=1}^{N-1} |A_{i+1} - A_i|}{\sum_{i=1}^{N} A_i} \]

RMS- Frame Energy

\[ \sqrt{\frac{1}{N} \sum_{n} |x(n)|^2} \]
2. Data

<table>
<thead>
<tr>
<th>None</th>
<th>Voice None</th>
<th>Voice</th>
<th>None</th>
<th>Voice</th>
<th>None</th>
<th>Voice</th>
<th>NoneVoice</th>
<th>NoneVoice</th>
<th>None</th>
<th>Voice</th>
<th>None</th>
<th>Voice</th>
</tr>
</thead>
</table>
2. Data

Power Spectral Density (PSD)

\[ \int_{t_1}^{t_2} \left| e^{-i2\pi ft} x(t) \right|^2 dt \]

Spectral Balance
(1 to 2kHz) vs (0 to 0.5kHz)
2. Data

Signal Intensity

$$10 \log_{10} \left( \frac{1}{4 \times 10^{-10}(t_2 - t_1)} \int_{t_1}^{t_2} x^2(t) \, dt \right)$$

Fundamental Frequency

Mel Frequency Cepstral Coefficients (MFCC)
We chose a subset of these: 

194 Selected Variables for each 5-Second Clip
3. Notion of “Best”

1. Gini Impurity
2. Adaboost Error Function
3. ROC AUC Score
3. Notion of “Best”: Gini Impurity

\[ G(k) = \sum_{i=1}^{J} P(i)(1 - P(i)) \]
3. Notion of “Best”: Adaboost Error Function

Adaboost Algorithm for Finding a Random Forest Model

AdaBoost Error Function

\[ \epsilon_m = \frac{\sum_{i=0}^{n} w_i^{(m)}(1 - \delta_{x,y})}{\sum_{i=0}^{n} w_i^{(m)}} \]
3. Notion of “Best”: ROC-AUC
Methods

1. Grid Search on hyperparameters
   a. Gini Impurity vs. Entropy
   b. Max. Depth of the Trees
   c. Learning Rate (0.25, 0.5, 0.75)

2. 5-Fold Cross Validation

3. Tools Used
   a. Praat
   b. Parselmouth
   c. Scikit-Learn
Results
Visualization of Last Decision Tree
# Most Important Features

<table>
<thead>
<tr>
<th>Importance</th>
<th>Audio Feature</th>
<th>Statistic</th>
<th>Clip Portion</th>
<th>Decrease in Gini Impurity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fundamental Frequency $f_0$</td>
<td>Median</td>
<td>Last 500 Milliseconds</td>
<td>0.384219106</td>
</tr>
<tr>
<td>2</td>
<td>Fundamental Frequency $f_0$</td>
<td>Minimum</td>
<td>Last 200 Milliseconds</td>
<td>0.38409663</td>
</tr>
<tr>
<td>3</td>
<td>Signal Intensity</td>
<td>Standard Deviation</td>
<td>Full Clip</td>
<td>0.288251239</td>
</tr>
<tr>
<td>4</td>
<td>Fundamental Frequency $f_0$</td>
<td>Standard Deviation</td>
<td>Last 500 Milliseconds</td>
<td>0.195265416</td>
</tr>
<tr>
<td>5</td>
<td>Fundamental Frequency $f_0$</td>
<td>% of Rising Slopes</td>
<td>Full Clip</td>
<td>0.059541905</td>
</tr>
<tr>
<td>6</td>
<td>6th MFCC</td>
<td>Mean</td>
<td>Last 500 Milliseconds</td>
<td>0.054253965</td>
</tr>
<tr>
<td>7</td>
<td>Signal Intensity</td>
<td>% of Rising Slopes</td>
<td>Last 200 Milliseconds</td>
<td>0.051263641</td>
</tr>
<tr>
<td>8</td>
<td>3rd MFCC</td>
<td>Standard Deviation</td>
<td>Full Clip</td>
<td>0.050303281</td>
</tr>
<tr>
<td>9</td>
<td>2nd MFCC</td>
<td>Mean</td>
<td>Full Clip</td>
<td>0.040921364</td>
</tr>
<tr>
<td>10</td>
<td>11th MFCC</td>
<td>Mean</td>
<td>Last 500 Milliseconds</td>
<td>0.0406494</td>
</tr>
<tr>
<td>11</td>
<td>Signal Intensity</td>
<td>Median</td>
<td>Last 500 Milliseconds</td>
<td>0.040223961</td>
</tr>
<tr>
<td>12</td>
<td>Fundamental Frequency $f_0$</td>
<td>Range</td>
<td>Last 500 Milliseconds</td>
<td>0.040138624</td>
</tr>
<tr>
<td>13</td>
<td>9th MFCC</td>
<td>Mean</td>
<td>Full Clip</td>
<td>0.027013746</td>
</tr>
<tr>
<td>14</td>
<td>Fundamental Frequency $f_0$</td>
<td>Slope of the First-Last Line</td>
<td>Full Clip</td>
<td>0.026454863</td>
</tr>
<tr>
<td>15</td>
<td>Fundamental Frequency $f_0$</td>
<td>Mean</td>
<td>Last 500 Milliseconds</td>
<td>0.020519628</td>
</tr>
<tr>
<td>16</td>
<td>12th MFCC</td>
<td>Mean</td>
<td>Last 500 Milliseconds</td>
<td>0.016024721</td>
</tr>
<tr>
<td>17</td>
<td>Fundamental Frequency $f_0$</td>
<td>Slope of the Least-Squares Line</td>
<td>Last 200 Milliseconds</td>
<td>0.015291915</td>
</tr>
</tbody>
</table>
Performance acc. to Notion of “Best”

On Test Sample of 4000 clips:

<table>
<thead>
<tr>
<th></th>
<th>Actual: “Question”</th>
<th>Actual: “Not a Question”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicted: “Question”</td>
<td>13 (True Positive)</td>
<td>4 (False Positive)</td>
</tr>
<tr>
<td>Predicted: “Not a Question”</td>
<td>452 (False Negative)</td>
<td>3531 (True Negative)</td>
</tr>
</tbody>
</table>

**ROC AUC Score:** 0.5134

**Overall Accuracy Score:** 88.6%

“Question” Acc. Score: 2.8%

“Not Question” Acc. Score: 99.9%
Future Work to Improve the Model

- Train on more even proportion of questions vs. non-questions
- Use all ~105,000 clips that we had (requires a lot of time)
- Use only most important features to make the model simpler
- Experiment with better data orientation techniques
- Run a larger combination of Grid Search hyperparameters
  - Ex: Clip Sizes, Clip Periods, Learning Rates, Max. Depth, Max. # of Features, etc.
- Combine with word content analysis already created by the MTC group
Thank You