The Music Information Retrieval Evaluation eXchange (MIREX)

MIREX DIY: A "Do-It-Yourself" Model for Future MIREX Tasks

http://music-ir.org/mirexwiki/
http://cluster3.lis.uiuc.edu/mirexdiydemo

J. Stephen Downie
Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign
jdownie@uiuc.edu
IMIRSEL

- International Music Information Retrieval Systems Evaluation Lab
- Faculty
  - J. Stephen Downie
- Graduate Students
  - Mert Bay, Andreas Ehmann, Anatoliy Gruzd, Xiao Hu, M. Cameron Jones, Jin Ha (Gina) Lee, Qin Wei, Kris West, Byunggil Yoo
- Undergraduates
  - Martin McCrory
IMIRSEL Model
MIREX Model

- Based upon the TREC approach:
  - Standardized queries/tasks
  - Standardized collections
  - Standardized evaluations of results
- Not like TREC with regard to distributing data collections to participants
  - Music copyright issues, ground-truth issues, overfitting issues
MIREX Overview

- Began in 2005
- Tasks defined by community debate
- Data sets collected and/or donated
- Participants submit code to IMIRSEL
- Code rarely works first try 😊
- Data collections tend to contain corrupt files
- Meet at ISMIR to discuss results
2006 General Statistics

- 13 tasks
- 46 teams
- 50 individuals
- 14 different countries
- 10 different programming languages and execution environments
- 92 individual runs
- 98 result data matrices on the wiki
MIREX 2006 Tasks

- Audio Beat Tracking
- Audio Cover Song Identification
- Audio Melody Extraction (2 subtasks)
- Audio Music Similarity and Retrieval
- Audio Onset Detection
- Audio Tempo Extraction
- Query-by-Singing or Humming (2 subtasks)
- Score Following
- Symbolic Melodic Similarity (3 subtasks)
MIREX 2006 Tasks

- Audio Beat Tracking
- Audio Cover Song Identification
- Audio Melody Extraction (2 subtasks)
- Audio Music Similarity and Retrieval
- Audio Onset Detection
- Audio Tempo Extraction
- Query-by-Singing or Humming (2 subtasks)
- Score Following
- Symbolic Melodic Similarity (3 subtasks)
New this year

- New Tasks
  - Audio Cover Song
  - Score Following
  - QBSH
- New Evaluations
  - Multiple parameters in Onset Detection
  - Evalutron 6000: Human similarity judgments
  - Friedman tests
Onset Detection
Evalutron 6000

<table>
<thead>
<tr>
<th></th>
<th>Audio Similarity</th>
<th>Symbolic Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td># Graders</td>
<td>24</td>
<td>21</td>
</tr>
<tr>
<td># Graders per Q/C pair</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td># Queries per grader</td>
<td>7-8</td>
<td>15</td>
</tr>
<tr>
<td>Size of Candidate lists</td>
<td>Max 30</td>
<td>15</td>
</tr>
<tr>
<td># Of Q/C pairs evaluated per grader</td>
<td>Max 240</td>
<td>225</td>
</tr>
<tr>
<td># Of queries</td>
<td>60</td>
<td>17</td>
</tr>
</tbody>
</table>
Friedman Tests

Audio Music Similarity and Retrieval

<table>
<thead>
<tr>
<th>Source</th>
<th>SS</th>
<th>df</th>
<th>MS</th>
<th>Chi-Sq</th>
<th>Prob&gt;Chi-Sq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Columns</td>
<td>84.733</td>
<td>5</td>
<td>16.947</td>
<td>24.291</td>
<td>0.000</td>
</tr>
<tr>
<td>Error</td>
<td>961.767</td>
<td>295</td>
<td>3.260</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>1046.50</td>
<td>359</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Mean Fine Score
Friedman’s Test: Multiple Comparisons

<table>
<thead>
<tr>
<th>TeamID</th>
<th>TeamID</th>
<th>Lowerbound</th>
<th>Mean</th>
<th>Upperbound</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>EP</td>
<td>TP</td>
<td>-0.963</td>
<td>0.008</td>
<td>0.980</td>
<td>FALSE</td>
</tr>
<tr>
<td>EP</td>
<td>VS</td>
<td>-0.755</td>
<td>0.217</td>
<td>1.188</td>
<td>FALSE</td>
</tr>
<tr>
<td>EP</td>
<td>LR</td>
<td>-0.630</td>
<td>0.342</td>
<td>1.313</td>
<td>FALSE</td>
</tr>
<tr>
<td>EP</td>
<td>KWT</td>
<td>-0.030</td>
<td>0.942</td>
<td>1.913</td>
<td>FALSE</td>
</tr>
<tr>
<td>EP</td>
<td>KWL</td>
<td>0.320</td>
<td>1.292</td>
<td>2.263</td>
<td>TRUE</td>
</tr>
<tr>
<td>TP</td>
<td>VS</td>
<td>-0.763</td>
<td>0.208</td>
<td>1.180</td>
<td>FALSE</td>
</tr>
<tr>
<td>TP</td>
<td>LR</td>
<td>-0.638</td>
<td>0.333</td>
<td>1.305</td>
<td>FALSE</td>
</tr>
<tr>
<td>TP</td>
<td>KWT</td>
<td>-0.038</td>
<td>0.933</td>
<td>1.905</td>
<td>FALSE</td>
</tr>
<tr>
<td>TP</td>
<td>KWL</td>
<td>0.312</td>
<td>1.283</td>
<td>2.255</td>
<td>TRUE</td>
</tr>
<tr>
<td>VS</td>
<td>LR</td>
<td>-0.847</td>
<td>0.125</td>
<td>1.097</td>
<td>FALSE</td>
</tr>
<tr>
<td>VS</td>
<td>KWT</td>
<td>-0.247</td>
<td>0.725</td>
<td>1.697</td>
<td>FALSE</td>
</tr>
<tr>
<td>VS</td>
<td>KWL</td>
<td>0.103</td>
<td>1.075</td>
<td>2.047</td>
<td>TRUE</td>
</tr>
<tr>
<td>LR</td>
<td>KWT</td>
<td>-0.372</td>
<td>0.600</td>
<td>1.572</td>
<td>FALSE</td>
</tr>
<tr>
<td>LR</td>
<td>KWL</td>
<td>-0.022</td>
<td>0.950</td>
<td>1.922</td>
<td>FALSE</td>
</tr>
<tr>
<td>KWT</td>
<td>KWL</td>
<td>-0.622</td>
<td>0.350</td>
<td>1.322</td>
<td>FALSE</td>
</tr>
</tbody>
</table>
IMIRSEL’s Future MIREX Plans

- Continue to explore more statistical significance testing procedures beyond Friedman’s ANOVA test
- Continue refinement of Evalutron 6000 technology and procedures
- Continue to establish a more formal organizational structure for future MIREX contests
- Continue to develop the evaluator software and establish an open-source evaluation API
- Make useful evaluation data publicly available year round
- Establish a webservice-based IMIRSEL/M2K online system prototype (Audio Cover Song Identification?)
Introducing...
But wait, it gets better......
The Amazing......
Music to Knowledge (M2K)

- Goal: Have both a toolset and the evaluation environment available to users
- Visual data flow programming built upon NCSA’s Data to Knowledge (D2K) machine learning environment
- Java-based, easily portable
- Supports distributed computing
How M2K/D2K Works

- Signal processing and machine learning code is written into *modules*
- Modules are ‘wired’ together to produce more complicated programs called *itineraries*
- Itineraries can then be run or used themselves as modules allowing *nesting* of programs
- Individual modules and nested itineraries can be assigned to be *parallelized* across all machines in a network, or to individual machines in a network
A Picture is Worth 1000 Words: Music Classifier Example
Music Classifier Example: Feature Extraction Nested Itinerary
Editing Parameters and Component Documentation
M2K: Main Goals

- Promote collaboration and sharing through a common, modular toolset
- A ‘black box’ approach to provide commonly needed algorithms for fast prototyping
- Alleviate the ‘reinventing the wheel’ problem
Example Consideration

- Music classification (artist, genre, etc) is often broken down into a feature extraction followed by a machine learning stage.
- Some researchers focus only on one stage or the other.
- Difficult to evaluate the success of approaches in this case.
- Ideally, would evaluate all feature extractors against all evaluators.
Integrating Other Tools

- Must also provide a means of support for all the other toolsets people use
  - MATLAB, Marsyas, Weka, Clam, ACE, and on and on
- External integration modules allow for non-M2K or JAVA-based programs to be used
  - E.g. C/C++ compiled binaries, MATLAB, etc
- External processes called through the Java runtime environment
M2K as an Evaluation Framework

- Ability to execute external programs allows M2K to be used as an execution controller, similar to a graphical scripting language
- Evaluation modules written to evaluate algorithms in various tasks
- M2K used for the in-house evaluation of both MIREX task sets (2005-2006)
  - Still not widely distributed
An External Classification Algorithm
Enter D2KWS

- D2K Web Service
- Built, again, by the ALG at NCSA
- Tomcat-based
- Has user/role permission model
- Some pleasantly magical features:
  - Automatic node assignment
  - Itineraries pretty much plug-and-play
Acknowledgements

- Special Thanks to:
  - *The Graduate School of Library and Information Science at the University of Illinois at Urbana-Champaign*
  - *The Andrew W. Mellon Foundation*
  - *The National Science Foundation (Grant No. NSF IIS-0327371)*
  - *Evalutron 6000 Graders*