Introduction: Awari

- Awari – ancient African game

- 4000 year old pit and pebble game
- Computers play Awari much better than humans
- Very fast search (Mpps, Bppm, 27-ply)
- Evaluation function is still very crude
Softwari

- Roel van der Goot
- Applet: web.cs.ualberta.ca/~games/

U of A wins the silver medal in London

Awari Endgame Databases

- Endgame databases are **BIG** (Chinook)
- Awari endgame DBs are **REALLY BIG**
- Provides an **oracle** of perfect knowledge, but nothing explicit about *general strategy*

Knowledge Discovery

- Incorporating human knowledge in a game playing program is always tough, but we *don’t even have* good domain knowledge for Awari
- Project Goals:
  - extract knowledge that can be used to construct a good evaluation function
  - add to human knowledge about Awari
Methods of Analysis

- A feature is a property (e.g. balance)
- Atomic feature: pit $i$ has $j$ pebbles
- Association rule mining, where each position is a “basket” of 12 items
- We have all possible transactions with weights (cf. Negative assoc. rules)
- Tally all values $\rightarrow$ expected value (EV)

Non-capture and capture positions
- data filtering

Results

Frequency of database values
EV visualization and frequency

Minimum frequency (Bayesian)

Frequency vs. number of pebbles

3-d visualization of EV-1 data
Minimum frequency effect
Minimum frequency effect

EV-1 (min freq. 100K)

EV-1 (overlay)
**EV-2 data:** pit_b = 2 pebbles

**Evaluation Function Results**

<table>
<thead>
<tr>
<th>result</th>
<th>frequency</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>lose</td>
<td>64694821</td>
<td>334</td>
</tr>
<tr>
<td>draw</td>
<td>22897612</td>
<td>118</td>
</tr>
<tr>
<td>win</td>
<td>105944287</td>
<td>547</td>
</tr>
<tr>
<td>sum</td>
<td>193536720</td>
<td>1000</td>
</tr>
<tr>
<td>net</td>
<td>293523543</td>
<td>+1.52</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>eval-2</th>
<th>lose</th>
<th>draw</th>
<th>win</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>lose</td>
<td>237</td>
<td>33</td>
<td>37</td>
<td>308</td>
</tr>
<tr>
<td>draw</td>
<td>37</td>
<td>20</td>
<td>32</td>
<td>90</td>
</tr>
<tr>
<td>win</td>
<td>59</td>
<td>63</td>
<td>477</td>
<td>600</td>
</tr>
<tr>
<td>sum</td>
<td>334</td>
<td>118</td>
<td>547</td>
<td>1000</td>
</tr>
</tbody>
</table>

**Future Work**

- Self-play games:
  - eval vs eval
  - eval vs database (count errors)
  - re-train on actual game positions
  - eval-1 on non-capture positions
- Other data mining techniques
Conclusions

- Successfully extracted knowledge from the Awari endgame database
- Built relatively good evaluation functions
- eval-1 and eval-2 also work well for unbalanced positions (e.g. diff=3), and can be further improved
- Discovered knowledge enhances human understanding of Awari