An Introduction to Data Profiling

27.4.2017
Felix Naumann
Overview

1. Profiling tasks
2. Profiling tools
3. Visualization
4. Next generation profiling
5. Profiling challenges
|    | A    | B     | C     | D     | E     | F     | G     | H     | I     | J     | K     | L     | M     | N     | O     | P     | Q     | R     | S     | T     | U     | V     | W     | X     | Y     | Z     | AA    | AB    | AC    |
|----|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| A1 | USA  | PONTE VEDRA BEACH FL | 201720005 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A2 | CANADA | TORONTO ON | 201720016 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A3 | UNITED KINGDOM | LONDON UK | 201720021 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A4 | AUSTRALIA | SYDNEY NSW | 201720026 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A5 | CHINA | BEIJING CHINA | 201720031 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A6 | JAPAN | TOKYO JAPAN | 201720036 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A7 | RUSSIA | MOSCOW RUSSIA | 201720041 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A8 | INDIA | DELHI INDIA | 201720046 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A9 | BRAZIL | SAO PAULO BRAZIL | 201720051 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A10| SOUTH AFRICA | JOHANNESBURG SOUTH AFRICA | 201720056 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |
| A11| SOUTH KOREA | SEOUL SOUTH KOREA | 201720061 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 567890 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 | 123456 | 789012 | 345678 | 901234 |

Data Profiling
Summer 2017

Felix Naumann
<table>
<thead>
<tr>
<th>county</th>
<th>county_desc</th>
<th>voter_reg_status</th>
<th>reason</th>
<th>state</th>
<th>zip_code</th>
<th>mail_addr</th>
<th>mail_addr</th>
<th>mail_city</th>
<th>mail_state</th>
<th>mail_zipcode</th>
<th>full_phone</th>
<th>race_code</th>
<th>ethnicity_code</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALAMANCE</td>
<td>908596 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27753</td>
<td>421 WHITFORD RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>PO BOX 4277</td>
<td>27753</td>
<td>229 1110</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>904372 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27753</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901674 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>229 8894</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>902598 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>229 8894</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>903125 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>229 8894</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>902137 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901208 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900202 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>909589 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>902135 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901564 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901978 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>901435 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>AARON</td>
<td>900143 A</td>
<td>ACTIVE</td>
<td>AV</td>
<td>NC</td>
<td>27215</td>
<td>513 EDST RD</td>
<td>BURLINGTON</td>
<td>27215</td>
<td>229 8894</td>
<td>27753</td>
<td>229 8894</td>
<td>W</td>
<td>UNA</td>
</tr>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>I</td>
<td>J</td>
<td>K</td>
<td>L</td>
<td>M</td>
<td>N</td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Number of rows: 1005
<table>
<thead>
<tr>
<th>State</th>
<th>Zipcode</th>
<th>Full Phone</th>
<th>Code</th>
<th>Race Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>North Carolina</td>
<td>27258</td>
<td>336 201.3354</td>
<td>W</td>
<td></td>
</tr>
<tr>
<td>New York</td>
<td>10010</td>
<td>212 212.2121</td>
<td>A</td>
<td></td>
</tr>
</tbody>
</table>

The table above shows a sample of data from a spreadsheet. The data includes states, zipcodes, full phone numbers, and race codes. The spreadsheet is being used for data profiling.
<table>
<thead>
<tr>
<th>voter_status</th>
<th>last_name</th>
<th>first_name</th>
<th>names_street_address</th>
<th>city</th>
<th>state</th>
<th>zip_code</th>
<th>mail_zipcode</th>
<th>full_phone</th>
<th>race_code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verified</td>
<td>ABLE</td>
<td>ELISA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verified</td>
<td>ABELE</td>
<td>KLAY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verified</td>
<td>ABELE</td>
<td>KLAY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verified</td>
<td>ABELE</td>
<td>MARY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verified</td>
<td>ABOBB</td>
<td>KLAY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verified</td>
<td>ABOBB</td>
<td>KLAY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verified</td>
<td>ABOBB</td>
<td>KLAY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verified</td>
<td>ABOBB</td>
<td>MARY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*NOTE: The table above is a screenshot of a spreadsheet with data about voters. The columns include voter status, last name, first name, names street address, city, state, zip code, mail zip code, full phone, and race code.*
<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>City</th>
<th>State</th>
<th>Phone</th>
<th>Gender</th>
<th>Birthdate</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOUNG</td>
<td>3207 STONE ST, MEbane</td>
<td>MEbane</td>
<td>NC</td>
<td>633-573</td>
<td>F</td>
<td>06/29/1990</td>
</tr>
<tr>
<td>YUQI</td>
<td>12216 RASPBERRY GRAHAM</td>
<td>GRAHAM</td>
<td>NC</td>
<td>72758</td>
<td>A</td>
<td>10/20/1990</td>
</tr>
<tr>
<td>OUYEN</td>
<td>552 MEADOWOYB, BURLINGTON</td>
<td>BURLINGTON</td>
<td>NC</td>
<td>51725</td>
<td>F</td>
<td>09/28/2012</td>
</tr>
<tr>
<td>ABA</td>
<td>12924 GRACE AVE, BURLINGTON</td>
<td>BURLINGTON</td>
<td>NC</td>
<td>72757</td>
<td>U</td>
<td>01/08/1990</td>
</tr>
<tr>
<td>YUNI</td>
<td>104 PHOENIX DR, ELON</td>
<td>ELON</td>
<td>NC</td>
<td>72744</td>
<td>UNA</td>
<td>05/08/2012</td>
</tr>
<tr>
<td>YUNI</td>
<td>736 BIRKDALE, MEbane</td>
<td>MEbane</td>
<td>NC</td>
<td>72757</td>
<td>UNA</td>
<td>05/08/2012</td>
</tr>
<tr>
<td>YUNI</td>
<td>3475 S NC HWY 11, HA W RIVER</td>
<td>HA W RIVER</td>
<td>NC</td>
<td>72758</td>
<td>U</td>
<td>06/30/2010</td>
</tr>
<tr>
<td>ZAMARRIPA</td>
<td>2028 PO BOX 791, MEbane</td>
<td>MEbane</td>
<td>NC</td>
<td>72757</td>
<td>U</td>
<td>09/20/2003</td>
</tr>
<tr>
<td>ZAMARRIPA</td>
<td>8090 PO BOX 791, MEbane</td>
<td>MEbane</td>
<td>NC</td>
<td>72757</td>
<td>U</td>
<td>09/20/2003</td>
</tr>
<tr>
<td>ZAMARRIPA</td>
<td>4090 PO BOX 791, MEbane</td>
<td>MEbane</td>
<td>NC</td>
<td>72757</td>
<td>U</td>
<td>09/20/2003</td>
</tr>
<tr>
<td>ZAMARRIPA</td>
<td>2028 PO BOX 791, MEbane</td>
<td>MEbane</td>
<td>NC</td>
<td>72757</td>
<td>U</td>
<td>09/20/2003</td>
</tr>
</tbody>
</table>

[Google Search: Excel histogram](https://www.google.com)
<table>
<thead>
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
<th>Column 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
<td>Value 4</td>
</tr>
<tr>
<td>Value 5</td>
<td>Value 6</td>
<td>Value 7</td>
<td>Value 8</td>
</tr>
<tr>
<td>Value 9</td>
<td>Value 10</td>
<td>Value 11</td>
<td>Value 12</td>
</tr>
<tr>
<td>Value 13</td>
<td>Value 14</td>
<td>Value 15</td>
<td>Value 16</td>
</tr>
</tbody>
</table>
Many interesting questions remain

What are possible keys and foreign keys?

- Phone
- firstname, lastname, street

Are there any functional dependencies?

- zip -> city
- race -> voting behavior

Which columns correlate?

- county and last name
- DoB and first name

What are frequent patterns in a column?

- ddddd
- dd aaa St
Data profiling is the process of examining the data available in an existing data source [...] and collecting statistics and information about that data.

- Wikipedia 09/2013

Data profiling refers to the activity of creating small but informative summaries of a database.

- Ted Johnson, Encyclopedia of Database Systems

A fixed set of data profiling tasks / results
Classification of Traditional Profiling Tasks

- **Single column**
  - Cardinalities
  - Patterns and data types
  - Value distributions

- **Multiple columns**
  - Uniqueness
    - Key discovery
    - Conditional
    - Partial
  - Inclusion dependencies
    - Foreign key discovery
    - Conditional
    - Partial
  - Functional dependencies
    - Conditional
    - Partial
Single-column vs. multi-column

- **Single column profiling**
  - Most basic form of data profiling
  - Assumption: All values are of same type
  - Assumption: All values have some common properties
    - That are to be discovered
  - Often part of the basic statistics gathered by DBMS
  - Complexity: Number of values/rows

- **Multicolumn profiling**
  - Discover joint properties
  - Discover dependencies
  - Complexity: Number of columns and number of values
Cardinalities

- Number of values
- Number of distinct values
- Number of NULLs
- MIN and MAX value

Useful for
- Query optimization
- Categorization of attribute
- Relevance of attribute
Numeric distributions

- Probability distribution for numeric values
- Detect whether data follows some well-known distribution
  - Determine that distribution function for data values
- If no specific/useful function detectable: histograms

**Normal distributions**

**Laplace distributions**
Determine (and display) value frequencies for value intervals
- Or for individual values
- Estimation of probability distribution for continuous variables

Useful for
- Query optimization
- Outlier detection
- Visualizing distribution
Data types and value patterns

- String vs. number
- String vs. number vs. date
- Categorical vs. continuous
- SQL data types
  - CHAR, INT, DECIMAL, TIMESTAMP, BIT, CLOB, ...
- Domains
  - VARCHAR(12) vs. VARCHAR (13)
- XML data types
  - More fine grained
- Regular expressions \( (\text{d}\{3\})-(\text{d}\{3\})-(\text{d}\{4\})-(\text{d}+) \)
- Semantic domains
  - Address, phone, email, first name

Increasing Difficulty
Uniqueness and keys

- **Unique column**
  - Only unique values
- **Unique column combination**
  - Only unique value combinations
  - Minimality: No subset is unique
- **(Primary) key candidate**
  - No null values
  - Uniqueness and non-null in one instance does not imply key: Only human can specify keys (and foreign keys)
- **Meaning of NULL values?**
- **Useful for**
  - Schema design, data integration, indexing, optimization
  - Inverse: non-uniques are duplicates
Inclusion dependencies and foreign keys

- $A \subseteq B$: All values in $A$ are also present in $B$
- $A_1, \ldots, A_i \subseteq B_1, \ldots, B_i$
  All value combinations in $A_1, \ldots, A_i$ are also present in $B_1, \ldots, B_i$

- Prerequisite for foreign key
  - Used across relations
  - Use across databases
  - But again: Discovery on a given instance, only user can specify for schema
Functional Dependencies

Game of Dependencies
Some Functional Dependencies:

1. Person $\rightarrow$ Lineage
2. Person $\rightarrow$ Hair
3. Person $\rightarrow$ Religion
4. Lineage $\rightarrow$ Hair
5. Religion, Hair $\rightarrow$ Lineage
6. ...

Ned Stark: "#4 looks like a reasonable quality constraint"

Ned Stark: "I believe Joffrey violates my database constraint."
Uses and Algorithms for FDs

- Schema design
  - Normalization
  - Keys
- Data cleansing
- Query optimization
- Key discovery

Naïve discovery approach
- For each column combination X
  - For each pair of tuples (t1,t2)
    - If t1[X\A] = t2[X\A] and t1[A] ≠ t2[A]: Break
- Exponential in number of attributes times number of rows squared
Partial dependencies

- INDs and FDs that do not perfectly hold
  - For all but 10 of the tuples
  - Only for 80% of the tuples
  - Only for 1% of the tuples

- Also for patterns, types, uniques, and other constraints

- Useful for
  - Data cleansing
Conditional dependencies

- Given a partial IND or FD: For **which** part do the hold?
- Expressed as a condition over the attributes of the relation
- Problems:
  - Infinite possibilities of conditions
  - Interestingness:
    - Many distinct values: less interesting
    - Few distinct values: surprising condition – high coverage
- Useful for
  - Integration: cross-source cINDs
Data profiling vs. data mining

- Data profiling gathers technical metadata to support data management.
- Data mining and data analytics discovers non-obvious results to support business management.

- Data profiling results: information about columns and column sets.
- Data mining results: information about rows or row sets.
  - clustering, summarization, association rules, ...

- Rahm and Do, 2000
  - Profiling: Individual attributes
  - Mining: Multiple attributes
Overview

1. Profiling tasks
2. **Profiling tools**
3. Visualization
4. Next generation profiling
5. Profiling challenges
Data profiling tools and algorithms

- IBM InfoSphere Information Analyzer

- Oracle Enterprise Data Quality

- Talend Data Quality

- Ataccama DQ Analyzer

- SAP BusinessObjects Data Insight and SAP BusinessObjects Information Steward

- Informatica Data Explorer

- Microsoft SQL Server Integration Services Data Profiling Task and Viewer

- Trillium Software Data Profiling

- CloverETL Data Profiler

- OpenRefine
  - [http://www.openrefine.org](http://www.openrefine.org)

- and many more…

Often packaged with data quality / data cleansing software
Very long feature lists

- Num rows
- Min value length
- Median value length
- Max value length
- Avg value length
- Precision of numeric values
- Scale of numeric values
- Quartiles
- Basic data types
- Num distinct values ("cardinality")
- Percentage null values
- Data class and data type
- Uniqueness and constancy
- Single-column frequency histogram
- Multi-column frequency histogram
- Pattern discovery (Aa9)
- Soundex frequencies
- Benford Law Frequency
- Single column primary key discovery
- Multi-column primary key discovery
- Single column IND discovery
- Inclusion percentage
- Single-column FK discovery
- Multi-column IND discovery
- Multi-column FK discovery
- Value overlap (cross domain analysis)
- Single-column FD discovery
- Multi-column FD discovery
- Text profiling
Statement about the distribution of first digits $d$ in (many) naturally occurring numbers:

- $P(d) = \log_{10}(d + 1) - \log_{10}(d) = \log_{10}(1 + \frac{1}{d})$
- Holds if $\log(x)$ is uniformly distributed

Picking a random $x$ position uniformly on this number line, roughly 30% of the time the first digit of the number will be 1.
Exponential growth
- Year 1: 1-100
- Year 2: 101-200
- Year 3: 201-400
- Year 4: 401-800
- Year 5: 801-1600
- Year 6: 3200

Function: \(2^{(year-1)} \cdot 100\)

First digit
- **1**: Some of year 1, all of year 2
- **2**: 7 months in year 3
  - \((\log_2(300/100)+1 - 2) \times 12\)
- **3**: remaining 5 months
- **4 – 9**: few months each
- **1**: again after under 4 months in year 5 for 1 whole year
  - \((\log_2(10)+1 - 4) \times 12\)
Examples for Benford’s Law

- Surface areas of 335 rivers
- Sizes of 3259 US populations
- 1800 molecular weights
- 5000 entries from a mathematical handbook
- 308 numbers in an issue of Reader's Digest
- Street addresses of the first 342 persons listed in American Men of Science

### Heights of the 60 tallest structures

<table>
<thead>
<tr>
<th>Leading digit</th>
<th>Count</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td>43.3%</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>11.7%</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>15.0%</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>10.0%</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>6.7%</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1.7%</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>3.3%</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>8.3%</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0.0%</td>
</tr>
</tbody>
</table>

- In Benford’s law
  - 30.1%
  - 17.6%
  - 12.5%
  - 9.7%
  - 7.9%
  - 6.7%
  - 5.8%
  - 5.1%
  - 4.6%

http://en.wikipedia.org/wiki/List_of_tallest_buildings_and_structures_in_the_world#Tallest_structure_by_category

Felix Naumann
Data Profiling
Summer 2017
Occurrences of leading digits in WikiTable numbers
An anonymous tool:
- "Cross-table analysis enables you to identify matching or orphaned records between two tables, based on a fully-customizable join condition and optional filter on either table."

Corresponds to
- `SELECT COUNT(*)`  
  FROM `A, B`  
  WHERE `A.x = B.y`  
  AND `cond`
- Plus some fancy visualization:

```
1700  
23
```
Screenshots from Talend Data Quality
### Simple Statistics

<table>
<thead>
<tr>
<th>Label</th>
<th>Count</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row Count</td>
<td>10,341</td>
<td>100.00%</td>
</tr>
<tr>
<td>Null Count</td>
<td>60.00</td>
<td>N/A</td>
</tr>
<tr>
<td>Distinct Count</td>
<td>10,275</td>
<td>N/A</td>
</tr>
<tr>
<td>Unique Count</td>
<td>10,273</td>
<td>N/A</td>
</tr>
<tr>
<td>Duplicate Count</td>
<td>2.00</td>
<td>N/A</td>
</tr>
<tr>
<td>Blank Count</td>
<td>8.00</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### Frequency Statistics

<table>
<thead>
<tr>
<th>value</th>
<th>count</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null field</td>
<td>60.00</td>
<td>N/A</td>
</tr>
<tr>
<td>Empty field</td>
<td>8.00</td>
<td>N/A</td>
</tr>
<tr>
<td><a href="mailto:JamesLowry@Oak.Bay.org">JamesLowry@Oak.Bay.org</a></td>
<td>1.00</td>
<td>N/A</td>
</tr>
<tr>
<td><a href="mailto:AnnDeborde@Bremerton.org">AnnDeborde@Bremerton.org</a></td>
<td>1.00</td>
<td>N/A</td>
</tr>
<tr>
<td>NancyPietro@Berkeley</td>
<td>1.00</td>
<td>N/A</td>
</tr>
<tr>
<td>RichardWellington@Santa Monica</td>
<td>1.00</td>
<td>N/A</td>
</tr>
<tr>
<td>JoyceBroderick@La Cruz.org</td>
<td>1.00</td>
<td>N/A</td>
</tr>
<tr>
<td>JeremyPawloski@Beverly Hills</td>
<td>1.00</td>
<td>N/A</td>
</tr>
<tr>
<td><a href="mailto:AnitaBarton@Burbank.org">AnitaBarton@Burbank.org</a></td>
<td>1.00</td>
<td>N/A</td>
</tr>
<tr>
<td>ElzaTopp@Santa Anita.org</td>
<td>1.00</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Screenshots for IBM Information Analyzer
Typical Shortcomings of Tools (and research methods)

- **Usability**
  - Complex to configure
  - Results complex to view and interpret

- **Scalability**
  - Main-memory based
  - SQL based DBMS

- **Efficiency**
  - Coffee, Lunch, Overnight

- **Functionality**
  - Restricted to simplest tasks
  - Restricted to individual columns or small column sets
    - “Realistic” key candidates vs. further use-cases
  - „Checking“ vs. „discovery“

- **Interpretation of profiling results**

**Why are DBMS a poor choice here?**

**That’s the big one**
Metanome
An extensible architecture

Backend
- Algorithm execution
- Result & Resource management

Frontend
- Algorithm configuration
- Result & Resource presentation

Metanome Store
- Configuration
- Resource Links
- Results

Input Sources
- MySQL
- DB2
- txt
- tsv
- xml
- csv

Profiling Algorithms
- BINDER jar
- SWAN jar
- SPIDER jar
- DUCC jar
- DFD jar
Overview

1. Profiling tasks
2. Profiling tools
3. **Visualization**
4. Next generation profiling
5. Profiling challenges
Motivation

- Human in the loop for data profiling and data cleansing.

- Advanced visualization techniques
  - Beyond bar-charts and pie-charts

- Interactive visualization
  - Support users in visualizing data, profiling results
  - Support any action taken upon the results
    - Cleansing, sorting, ...
    - Re-profile and visualize immediately

Further reading:
http://vgc.poly.edu/~juliana/courses/cs9223/Lectures/intro-to-visualization.pdf
Profiler: Integrated Statistical Analysis and Visualization for Data Quality Assessment

http://vis.stanford.edu/files/2012-Profiler-AVI.pdf

Figure 1: The Profiler User Interface. The UI contains (clockwise from top-left): (a) linked summary visualizations, and (d) anomaly browser. Profiler generates a set of linked summary visualizations, and we investigate possible causes of missing MPAA movie ratings. The grey bar above we select it to highlight matching records. The Release Date chart shows that missing ratings correlate with earlier release dates.

Figure 6: Taxonomy of Data Quality Issues. We list classes of methods for detecting each issue, example routines used in Profiler, and visualizations for assessing their output.
Massive screens for massive data

- Powerwall, University Konstanz
- 5.20 m x 2.15 m; almost nine million pixels

https://www.vis.uni-konstanz.de/forschung/powerwall/
Overview

1. Profiling tasks
2. Profiling tools
3. Visualization
4. **Next generation profiling**
5. Profiling challenges
Extended Classification of Profiling Tasks

Data Profiling

Single source

Single column

Cardinalities

Patterns and data types

Uniqueness and keys

Distributions

Uniqueness and keys

Inclusion and foreign key dep.

Functional dependencies

Conditional and approximate dep.

Multiple columns

Duplicate detection

Record linkage

Data overlap

Schema matching

Schematic overlap

Cross-schema dependencies

Multiple sources

Topical overlap

Topic discovery

Topical clustering

Schematic overlap

Record linkage

Duplicate detection

Conditional and approximate dep.

Data overlap

Topics discovery

Topical clustering

Multiple columns

Functional dependencies

Inclusion and foreign key dep.

Patterns and data types

Uniqueness and keys

Cardinalities

Single source

Uniqueness and keys

Distributions

Functional dependencies

Inclusion and foreign key dep.
(Semi-)Automatically determine cross-schema value correspondences between attributes

Traditionally: Input for data transformation and exchange tasks

With data profiling
- Extract features from schema and data
  - Instance-based schema matching
- Compare features
  - Similarity measure, machine learning

For data profiling
- Determine closeness of two schemata
- Determine „schema fit“
  - Complement (few matches) or union (many matches)
Inclusion dependencies across schemata

- Join paths between data sources

Conditional INDs

- Typical pattern among crossreferencing sources

---

**Schema Complement:**

**Cross-Schema Dependencies**

<table>
<thead>
<tr>
<th>Unit cost</th>
<th>DBName</th>
<th>ProdID</th>
</tr>
</thead>
<tbody>
<tr>
<td>200 USD</td>
<td>ToyDB</td>
<td>17</td>
</tr>
<tr>
<td>50 EUR</td>
<td>ToyDB</td>
<td>18</td>
</tr>
<tr>
<td>1000 QAR</td>
<td>FashionDB</td>
<td>18</td>
</tr>
</tbody>
</table>

**Catalog**

**ToyDB**

<table>
<thead>
<tr>
<th>EntityID</th>
<th>further data</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>abcd...</td>
</tr>
<tr>
<td>18</td>
<td>efg...</td>
</tr>
</tbody>
</table>

**FashionDB**

<table>
<thead>
<tr>
<th>EntityID</th>
<th>further data</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>abcd...</td>
</tr>
<tr>
<td>19</td>
<td>efg...</td>
</tr>
</tbody>
</table>
Duplicate Detection and Record Linkage

- Detect multiple (different) representations of the same real-world entity

- Traditionally: Input for data cleansing and data fusion tasks
  - See second part of course

- For data profiling
  - Intra-source: Determine duplicity/cleanliness
  - Inter-source: Determine „data fit“
    - Complement (few duplicates) or union (many duplicates)
Together: Profiling for Integration

- Create measures to estimate integration (and cleansing) effort
  - Schema and data overlap
  - Severity of heterogeneity
- Schema matching/mapping
  - What constitutes the “difficulty” of matching/mapping?
- Duplicate detection
  - Estimate data overlap
  - Estimate fusion effort

- Overall: Determine integration complexity and integration effort
  - Intrinsic complexity: Schema and data
  - Extrinsic complexity: Tools and expertise
Integration effort estimation

Integration specialist

Schema analysis (matching)

Data analysis (profiling)

Complexity model

Effort model

Integration tools (mapping & ETL)

Integration result

Measured effort

Estimated effort

Tool capabilities

User expertise

Production side

Estimation side

Measured effort

 Estimated effort
Topic discovery and clustering

- What is a data set about?
  - Domain(s), topics, entity types

- Single-topic datasets
  - GraceDB, IMDB, Sports databases, etc.

- Multi-topic datasets
  - General purpose knowledge bases: YAGO, DBpedia
  - 154 million tables crawled from the Web (2008)

- Topical clustering for
  - Source selection, query processing

- What is a topic?
  - Wikipedia categories?
Overview

1. Profiling tasks
2. Profiling tools
3. Visualization
4. Next generation profiling
5. Profiling challenges
Profiling Challenges

- Efficient profiling
- Scalable profiling
- Holistic profiling
- Incremental profiling
- Online profiling
- Profiling query results
- Profiling new types of data
- Data generation and testing
- Data profiling benchmark

Each of these is worth one (or more) master's theses!
Efficient and Scalable profiling

- Scalable in number of rows
- Scalable in number of columns
  - Number of column combinations is exponential
  - Small table with 100 columns: $2^{100} - 1 = 1,267,650,600,228,229,401,496,703,205,375$
    - $= 1.3$ nonillion combinations
    - Impossible to check
    - Impossible even to enumerate

- Solutions
  - Scale up: More RAM, faster CPUs
    - Expensive
  - Scale in: More cores
    - More complex (threads)
  - Scale out: More machines
    - Communication overhead
Holistic Profiling

- Various profiling methods for various profiling tasks
- Commonalities/similarities
  - Search space: All column combinations (or pairs thereof)
  - I/O: Read all data at least once
  - Data structure: Some index or hash table
  - Pruning and candidate generation: based on subset/superset relationships
  - Sortation: Benefit from sorted sets
- Challenge: Develop single method to output all/most profiling results
- Masterprojekt – paper submission
Incremental profiling

- Data is dynamic
  - Insert (batch or tuple-based)
    - Data streams
  - Updates
  - Deletes

- Problem: Keep profiling results up-to-date without reprofiling the entire data set
  - Easy examples: SUM, COUNT, AVG
  - Difficult examples: MIN, MAX, MEDIAN, uniqueness, FDs, etc.
DynFD – Masterproject 2016/17

1. Batch
2. Positive Cover
3. Negative Cover

Database
PLIs
Dictionary-encoded Records

Minimal FDs
Online Profiling

- Profiling is long procedure
  - Boring for developers
  - Expensive for machines (I/O and CPU)

- Challenge: Display intermediate results
  - ... of improving/converging accuracy
  - Allows early abort of profiling run

- Gear algorithms toward that goal
  - Allow intermediate output
  - Enable early output: "progressive" profiling

---

Felix Naumann
Data Profiling
Summer 2017
Query results are boring: Spruce them up with some metadata
- Usually only: Row count
- For each column, give some statistics
  - Uniqueness, histogram, AVG, etc.
  - Show FDs

Idea: Piggy-back profiling on query execution
- Re-use sortations, hash tables, etc.
Masterproject “Piggyback Profiling” (2013/14)

<table>
<thead>
<tr>
<th>rec.id</th>
<th>rec.name</th>
<th>ldiff</th>
<th>tr.id</th>
<th>tr.name</th>
<th>tr.position</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>She loves you</td>
<td>0</td>
<td>250</td>
<td>Check</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>Hig</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Cali</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Diagram showing runtime in seconds for different queries and profiles]

Felix Naumann
Data Profiling
Summer 2017
Profiling New Types of Data

- Traditional data profiling: Single table or multiple tables
- More and more data in other models
  - XML / nested relational / JSON
  - RDF triples
  - Textual data: Blogs, Tweets, News
  - Multimedia data
- Different models offer new dimensions to profile
  - XML: Nestedness, measures at different nesting levels
  - RDF: Graph structure, in- and outdegrees
  - Multimedia: Color, video-length, volume, etc.
  - Text: Sentiment, sentence structure, complexity, and other linguistic measures
Example: Text profiling

- **Statistical measures**
  - Syllables per word
  - Sentence length
  - Proportions of parts of speech

- **Vocabulary measures**
  - Frequencies of specific words
  - Type-token ratio
  - Simpson’s index (vocabulary richness)
  - Number of hapax (dis)legomena
    - Token that occurs exactly once (twice) in the corpus
    - Characterize style of an author

Idea and following figures based on:
"Literature Fingerprinting: A New Method for Visual Literary Analysis" by Daniel A. Keim and Daniela Oelke
Hapax Legomena

“Literature Fingerprinting: A New Method for Visual Literary Analysis” by Daniel A. Keim and Daniela Oelke
Verse length
News Statistics

Number of Sentences per Paragraph (as average per Text)

Counts (Percentage)

News CNN  News NYT  News WSJ

Master Thesis Matthias Kohnen
Summary

Data Profiling

Single source
- Single column
  - Cardinalities
  - Uniqueness and keys
  - Patterns and data types
  - Distributions
- Multiple columns
  - Uniqueness and keys
  - Inclusion and foreign key dep.
  - Functional dependencies
  - Conditional and approximate dep.

Multiple sources
- Topical overlap
  - Topic discovery
  - Topical clustering
- Schematic overlap
  - Schema matching
  - Cross-schema dependencies
- Data overlap
  - Duplicate detection
  - Record linkage