Uncovering Plagiarism, Authorship, and Social Software Misuse

Bauhaus-Universität Weimar
Martin Potthast, Tim Gollub, Matthias Hagen, Anna Beyer, Matthias Busse, Martin Tippmann, Johannes Kiesel, and Benno Stein

Universitat Politècnica de València
Parth Gupta and Paolo Rosso

University of the Aegean
Efstathios Stamatatos

Autoritas Consulting
Francisco Rangel

Duquesne University
Patrick Juola

Bar-Ilan University
Moshe Koppel

University of Lugano
Giacomo Inches

Illinois Institute of Technology
Shlomo Argamon
Uncovering Plagiarism, Authorship, and Social Software Misuse

Outline

- Author Profiling
- Author Identification
- Plagiarism Detection
- Software Submissions
- Summary
The PAN Competition

PAN is a network around digital text forensics.

Mission

- Foster research and development in our tasks
- Push the limits of evaluating them
- Improve methodology for lab-style evaluations

Tasks

- Author Profiling (new in 2013)
- Author Identification
- Plagiarism Detection

Software Submissions

- Instead of run submissions (i.e., software output on a given input)
- Improves sustainability, replicability, and reproducibility
- Increases participant engagement
- Allows for cross-year evaluations
Author Profiling

- Given a document, what are its author’s demographics?

Corpus

- Genre: social media
- Languages: English, Spanish
- Size: 346,100 authors
- Annotations: age, gender

Selected results

- 21 softwares submitted
- Gender difficult to be discriminated, somewhat better in Spanish
- Age correctly detected in about 2/3 of cases

Award from the ForensicLab of the Universitat Pompeu Fabra
the beginning of a time
itself didn’t matter any-
object was simply an
good-looking authors
students who would lis-
thinking, I could do

Image source: Jason Michael (http://www.flickr.com/photos/jasonmichael/309773150)
Author Identification

- Given a document, who wrote it?

Corpus

- Genres: non-fiction writing, short fiction, news
- Languages: English, Spanish, Greek
- Size: 120 cases
- Annotations: authorship

Selected results

- 18 softwares submitted
- Greek more difficult than English and Spanish
- Balancing performance in all languages with a single approach difficult
- Meta-model competitive to participants, but does not dominate
Plagiarism Detection

Given a document, is it an original?

Corpus

- Genre: web, news
- Language: English
- Size: 10000 suspicious documents
- Annotations: reused text passages, obfuscation

Selected results

- 19 softwares submitted
- Advanced evaluation framework for web-scale retrieval
- Different retrieval paradigms open up trade-off between costs and recall
- Summary plagiarism most difficult to be detected
- First-time cross-year evaluation; first steps toward all-time evaluation
Software Submissions
Software Submissions
Challenges ➔ Approaches

1. Environment diversity ➔ virtualization
   Support a wide variety of programming languages and operating systems.

2. Executing untrusted software ➔ virtualization
   Better be safe than sorry when executing binaries from a third party.

3. Data leakage ➔ sandboxing
   Prevent data leaking by running software in a secured environment.

4. Error handling ➔ unit testing
   Streamline the development round-trips for fixing execution errors.

5. Responsibility ➔ staged submissions
   Incentivice participants to submit early.

6. Execution cost ➔ provide hardware or raise usage fees
   We provided four servers each hosting up to 20 virtual machines.
Software Submissions

The 2013 Experience

- Entire lab accepts software submissions
- 62 virtual machines requested and provisioned
- 47 softwares installed, prepared for execution, and submitted by participants
- Testing and round-trips to fix errors
- Managed execution and evaluation using TIRA

The 2012 Experience

- One task accepts software submissions
- 10 softwares submitted
- Manual preparation for execution by us
- Testing and round-trips to fix errors
- Managed execution and evaluation using TIRA
Software Submissions

Error Analysis

Problems

- Input 13
- Error 118
- Output 32

<table>
<thead>
<tr>
<th>Format (Encoding)</th>
<th>Extraneous / Missing Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Extraneous / Missing Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
</tr>
<tr>
<td>14</td>
</tr>
</tbody>
</table>

Solutions

- Validation
- Corpus reorganization

- Validation
- Validation

- Validation

- 1493 mails exchanged in 392 conversations
- 39 of 46 teams experienced at least one error, 26 at least two, 1 team 10
- No one panicked
- Staged submissions helped resolve errors early on
- Rigorous unit testing and tools to assist participants in development
### Software Submissions

#### Error Analysis

<table>
<thead>
<tr>
<th>Problems</th>
<th>Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input 13</td>
<td>Validation</td>
</tr>
<tr>
<td>Execution 60</td>
<td>Corpus reorganization</td>
</tr>
<tr>
<td>Virtual Machine 13</td>
<td>Staged execution tests (increasing corpus size)</td>
</tr>
<tr>
<td>Output 32</td>
<td>Execution tests (parameter variation)</td>
</tr>
</tbody>
</table>

#### Problems

- **Input**
  - Format (Encoding) 8
  - Extraneous / Missing Information 5
- **Execution**
  - Code (Runtime / Resource Exception) 27
  - Parameter (Disregarded, Missing) 12
  - StdOut (Progress, Verbosity, Prompt) 8
  - API (Misuse), Environment, File Access 6
- **Virtual Machine**
  - Locked out 6
  - Insufficient Disk / RAM 7
- **Output**
  - Format 18
  - Extraneous / Missing Information 14

#### Solutions

- Validation
- Corpus reorganization
- Staged execution tests (increasing corpus size)
- Execution tests (parameter variation)
- Output parameters (quiet, progress, verbose), output format validation, output filtering
- Environment checks, execution tests
- Monitoring, health checks, access checks
- Resource request form
- Validation
- Validation

- 1493 mails exchanged in 392 conversations
- 39 of 46 teams experienced at least one error, 26 at least two, 1 team 10
- No one panicked
- Staged submissions helped resolve errors early on

→ Rigorous unit testing and tools to assist participants in development
## Software Submissions
### Cross-year Evaluation 2011-2013

<table>
<thead>
<tr>
<th>Software Submission</th>
<th>PlagDet on PAN Plagiarism Corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Year</td>
</tr>
<tr>
<td>Kong</td>
<td>2012</td>
</tr>
<tr>
<td>Oberreuter</td>
<td>2012</td>
</tr>
<tr>
<td>R. Torrejón</td>
<td>2013</td>
</tr>
<tr>
<td>Kong</td>
<td>2013</td>
</tr>
<tr>
<td>Palkovskii</td>
<td>2012</td>
</tr>
<tr>
<td>R. Torrejón</td>
<td>2012</td>
</tr>
<tr>
<td>Suchomel</td>
<td>2013</td>
</tr>
<tr>
<td>Suchomel</td>
<td>2012</td>
</tr>
<tr>
<td>Saremi</td>
<td>2013</td>
</tr>
<tr>
<td>Shrestha</td>
<td>2013</td>
</tr>
<tr>
<td>Kueppers</td>
<td>2012</td>
</tr>
<tr>
<td>Palkovskii</td>
<td>2013</td>
</tr>
<tr>
<td>Nourian</td>
<td>2013</td>
</tr>
<tr>
<td>Sánchez-Vega</td>
<td>2012</td>
</tr>
<tr>
<td>Baseline</td>
<td></td>
</tr>
<tr>
<td>Gillam</td>
<td>2012</td>
</tr>
<tr>
<td>Gillam</td>
<td>2013</td>
</tr>
<tr>
<td>Jayapal</td>
<td>2013</td>
</tr>
<tr>
<td>Jayapal</td>
<td>2012</td>
</tr>
</tbody>
</table>
Assessing corpus difficulty

![Graph showing PlagDet scores for PAN'11, PAN'12, and PAN'13 corpora across participants.](image)
Software Submissions
Cross-year Evaluation 2011-2013 (continued)

Assessing improvements across versions

<table>
<thead>
<tr>
<th>Participant</th>
<th>PAN'13 corpus</th>
<th>PAN'12 corpus</th>
<th>PAN'11 corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-0.4</td>
<td>-0.3</td>
<td>-0.2</td>
</tr>
<tr>
<td></td>
<td>-0.1</td>
<td>0</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0.2</td>
<td>0</td>
</tr>
</tbody>
</table>

PlagDet delta from 2012 to 2013
Summary

<table>
<thead>
<tr>
<th>Statistics</th>
<th>ALLC</th>
<th>SEPLN</th>
<th>FIRE</th>
<th>CLEF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task(s)</td>
<td>1</td>
<td>1</td>
<td>1 1 1</td>
<td>2 3 3 3</td>
</tr>
<tr>
<td>Follower</td>
<td></td>
<td>78</td>
<td>151 181 232 286</td>
<td></td>
</tr>
<tr>
<td>Registrations</td>
<td>11</td>
<td>21</td>
<td>6 12 16</td>
<td>53 52 68 110</td>
</tr>
<tr>
<td>Runs/Software</td>
<td>13</td>
<td>14</td>
<td>6 8 8</td>
<td>27 27 48 58</td>
</tr>
<tr>
<td>Notebooks</td>
<td>8</td>
<td>11</td>
<td>6 2 6</td>
<td>22 22 34 47</td>
</tr>
<tr>
<td>Attendees</td>
<td>5</td>
<td>18</td>
<td>6 30</td>
<td>25 36 61</td>
</tr>
</tbody>
</table>

Take-away messages

- Software submissions improve sustainability
- Software submissions allow for re-evaluation
- Software submissions allow for cross-year evaluation
- Software submissions do not discourage participation
## Summary

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Task(s)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Follower</td>
<td></td>
<td>78</td>
<td></td>
<td></td>
<td></td>
<td>151</td>
<td>181</td>
<td>232</td>
<td>286</td>
</tr>
<tr>
<td>Registrations</td>
<td>11</td>
<td>21</td>
<td>6</td>
<td>12</td>
<td>16</td>
<td>53</td>
<td>52</td>
<td>68</td>
<td>110</td>
</tr>
<tr>
<td>Runs/Software</td>
<td>13</td>
<td>14</td>
<td>6</td>
<td>8</td>
<td>8</td>
<td>27</td>
<td>27</td>
<td>48</td>
<td>58</td>
</tr>
<tr>
<td>Notebooks</td>
<td>8</td>
<td>11</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>22</td>
<td>22</td>
<td>34</td>
<td>47</td>
</tr>
<tr>
<td>Attendees</td>
<td>5</td>
<td>18</td>
<td>6</td>
<td>30</td>
<td></td>
<td>25</td>
<td>36</td>
<td>61</td>
<td></td>
</tr>
</tbody>
</table>

### Take-away messages

- Software submissions improve sustainability
- Software submissions allow for re-evaluation
- Software submissions allow for cross-year evaluation
- Software submissions do not discourage participation

Thank you for your attention!