Overview of the 6th International Competition on Plagiarism Detection

Martin Potthast, Matthias Hagen, Anna Beyer, Matthias Busse, Martin Tippmann, and Benno Stein
Bauhaus-Universität Weimar
www.webis.de

Paolo Rosso
Universitat Politècnica de València
Outline

- Introduction
- Source Retrieval
- Text Alignment
- Summary
**Plagiarism Detection**

**Source Retrieval**

**Given**
- suspicious document
- web search engine

**Task**
- retrieve plagiarized sources
- minimize retrieval costs

**Text Alignment**

**Given**
- pair of documents

**Task**
- extract passages of reused text

---

(Visual diagram of the process)

- Document collection
- Source retrieval
- Candidate documents
- Text alignment
- Knowledge-based post-processing
- Suspicious passages

---

©www.webis.de 2014
Source Retrieval

Participant

Web search

Web

Evaluation corpus

Corpus construction

Search proxy API

Performance measures and source oracle

Evaluation run

Submitted plagiarism detector

Virtual machines

Static evaluation infrastructure

TIRA experimentation platform

ChatNoir Cluster

ClueWeb09 (0.5 billion English web pages)
Source Retrieval

Corpus & Performance Measures

Corpus

- Webis Text Reuse Corpus 2012
- 297 manually written essay-length documents
- Each consists of up to 70 passages of reused text from the ClueWeb
- Training: 98 documents
- Test: 99 documents

Retrieval performance measures

- Precision, recall, and $F_\alpha$

Cost-effectiveness measures

- Workload as counts of queries and downloads
- Workload until 1st detection
- Runtime
An analysis of the participants’ notebooks reveals a source retrieval process:

1. **Chunking**
   Given a suspicious document, it is divided into (possibly overlapping) passages of text. Each chunk of text is then processed individually.

2. **Keyphrase Extraction**
   Given a chunk (or the entire suspicious document), keyphrases are extracted from it in order to formulate queries with them.

3. **Query Formulation**
   Given sets of keywords extracted from chunks, queries are formulated which are tailored to the API of the search engine used.

4. **Search Control**
   Given a set of queries, the search controller schedules their submission to the search engine and directs the download of search results.

5. **Download Filtering**
   Given a set of downloaded documents, all documents are removed that are not worthwhile for detailed comparison to the suspicious document.
## Source Retrieval

### Evaluation Results

<table>
<thead>
<tr>
<th>Team</th>
<th>Year</th>
<th>Downloaded Sources</th>
<th>Total Workload</th>
<th>Workload to 1st Detection Detect.</th>
<th>No.</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>F\textsubscript{1}</td>
<td>Prec.</td>
<td>Rec.</td>
<td>Queries</td>
<td>Dwlds</td>
</tr>
<tr>
<td>Elizalde</td>
<td>2013</td>
<td>0.16</td>
<td>0.12</td>
<td>0.37</td>
<td>41.6</td>
<td>83.9</td>
</tr>
<tr>
<td>Elizalde</td>
<td>2014</td>
<td>0.34</td>
<td>0.40</td>
<td>0.39</td>
<td>54.5</td>
<td>33.2</td>
</tr>
<tr>
<td>Foltynek</td>
<td>2013</td>
<td>0.11</td>
<td>0.08</td>
<td>0.26</td>
<td>166.8</td>
<td>72.7</td>
</tr>
<tr>
<td>Gillam</td>
<td>2013</td>
<td>0.06</td>
<td>0.04</td>
<td>0.15</td>
<td>15.7</td>
<td>86.8</td>
</tr>
<tr>
<td>Haggag</td>
<td>2013</td>
<td>0.38</td>
<td>0.67</td>
<td>0.31</td>
<td>41.7</td>
<td>5.2</td>
</tr>
<tr>
<td>Kong</td>
<td>2013</td>
<td>0.01</td>
<td>0.01</td>
<td>0.59</td>
<td>47.9</td>
<td>5185.3</td>
</tr>
<tr>
<td>Kong</td>
<td>2014</td>
<td>0.12</td>
<td>0.08</td>
<td>0.48</td>
<td>83.5</td>
<td>207.1</td>
</tr>
<tr>
<td>Lee</td>
<td>2013</td>
<td>0.40</td>
<td>0.58</td>
<td>0.37</td>
<td>48.4</td>
<td>10.9</td>
</tr>
<tr>
<td>Prakash</td>
<td>2014</td>
<td>0.39</td>
<td>0.38</td>
<td>0.51</td>
<td>60.0</td>
<td>38.8</td>
</tr>
<tr>
<td>Suchomel</td>
<td>2013</td>
<td>0.05</td>
<td>0.04</td>
<td>0.23</td>
<td>17.8</td>
<td>283.0</td>
</tr>
<tr>
<td>Suchomel</td>
<td>2014</td>
<td>0.11</td>
<td>0.08</td>
<td>0.40</td>
<td>19.5</td>
<td>237.3</td>
</tr>
<tr>
<td>Williams</td>
<td>2013</td>
<td>0.47</td>
<td>0.60</td>
<td>0.47</td>
<td>117.1</td>
<td>12.4</td>
</tr>
<tr>
<td>Williams</td>
<td>2014</td>
<td>0.47</td>
<td>0.57</td>
<td>0.48</td>
<td>117.1</td>
<td>14.4</td>
</tr>
<tr>
<td>Zubarev</td>
<td>2014</td>
<td>0.45</td>
<td>0.54</td>
<td>0.45</td>
<td>37.0</td>
<td>18.6</td>
</tr>
</tbody>
</table>

- Ranked by recall, the 2014 approaches outperform all except two from 2013
- Ensemble recall: 0.85; only 14 topic with ensemble recall less than 0.6
- Some returning participants improve (Elizalde, Suchomel)
Text Alignment
Text Alignment
Corpus & Performance Measures

Corpus

- The evaluation corpus has been reused from last year
- A supplemental corpus serves as baseline
- Problems / Criticism of “corpus reuse”
  - Gives rise to overfitting
  - Some participants found out about it
  ➔ In the future, we’ll be more open about this

Performance measures

- Plagdet, precision, recall, granularity, and runtime as usual
- New measures that capture more abstract aspects of detection performance
An analysis of the participants’ notebooks reveals a detailed comparison process:

1. **Seeding**
   Given a suspicious document and a source document, matches (also called “seeds”) between the two documents are identified using some seed heuristic. Seed heuristics either identify exact matches or create matches by changing the underlying texts in a domain-specific or linguistically motivated way.

2. **Extension**
   Given seed matches identified between a suspicious document and a source document, they are merged into aligned text passages of maximal length between the two documents which are then reported as plagiarism detections.

3. **Filtering**
   Given a set of aligned passages, a passage filter removes all aligned passages that do not meet certain criteria.
New trend: obfuscation prediction

Given a pair of documents, predict the most likely type of obfuscation of reused passages between them.

Approaches

- Decide a priori, before aligning the documents, which alignment strategy/parameters to apply
- Decide a posteriori, after aligning the documents using multiple alignment strategies/parameters, which result is best
- A priori decisions are machine-learning based
- A posteriori decisions are rule-based

Classification schemes

- no obfuscation vs. rest
- summaries vs. rest
- no obfuscation, random, summaries, rest
## Text Alignment

### Evaluation Results

<table>
<thead>
<tr>
<th>Team</th>
<th>PlagDet</th>
<th>Recall</th>
<th>Precision</th>
<th>Granularity</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sanchez-Perez</td>
<td>0.88</td>
<td>0.88</td>
<td>0.88</td>
<td>1.00</td>
<td>00:25:35</td>
</tr>
<tr>
<td>Oberreuter</td>
<td>0.87</td>
<td>0.86</td>
<td>0.89</td>
<td>1.00</td>
<td>00:05:31</td>
</tr>
<tr>
<td>Palkovskii</td>
<td>0.87</td>
<td>0.83</td>
<td>0.92</td>
<td>1.01</td>
<td>01:10:04</td>
</tr>
<tr>
<td>Glinos</td>
<td>0.86</td>
<td>0.79</td>
<td>0.96</td>
<td>1.02</td>
<td>00:23:13</td>
</tr>
<tr>
<td>Shrestha</td>
<td>0.84</td>
<td>0.84</td>
<td>0.86</td>
<td>1.01</td>
<td>69:51:15</td>
</tr>
<tr>
<td>R. Torrejón</td>
<td>0.83</td>
<td>0.77</td>
<td>0.90</td>
<td>1.00</td>
<td>00:00:42</td>
</tr>
<tr>
<td>Gross</td>
<td>0.83</td>
<td>0.77</td>
<td>0.93</td>
<td>1.03</td>
<td>00:03:00</td>
</tr>
<tr>
<td>Kong</td>
<td>0.82</td>
<td>0.81</td>
<td>0.84</td>
<td>1.00</td>
<td>00:05:26</td>
</tr>
<tr>
<td>Abnar</td>
<td>0.67</td>
<td>0.61</td>
<td>0.77</td>
<td>1.02</td>
<td>01:27:00</td>
</tr>
<tr>
<td>Alvi</td>
<td>0.66</td>
<td>0.55</td>
<td>0.93</td>
<td>1.07</td>
<td>00:04:57</td>
</tr>
<tr>
<td>Baseline</td>
<td>0.42</td>
<td>0.34</td>
<td>0.93</td>
<td>1.28</td>
<td>00:30:30</td>
</tr>
<tr>
<td>Gillam</td>
<td>0.28</td>
<td>0.17</td>
<td>0.87</td>
<td>1.00</td>
<td>00:00:55</td>
</tr>
</tbody>
</table>

- The top performers are Sanchez-Perez, Oberreuter, and Palkovskii
- Performances are very close together; further improvements may be difficult
- Summary obfuscation still most difficult; Glinos outperforms Sanchez-Perez

- PlagDet combines recall, precision, and granularity
- Granularity measures the number of times a plagiarism case is detected
Text Alignment
Performance Measures Revisited
Text Alignment

Performance Measures Revisited

- Currently, detection performance is measured at character level
- We introduce two more abstract levels: case level, and document level
- The new measures build upon the character level ones
Text Alignment

Performance Measures Revisited

- Currently, detection performance is measured at character level
- We introduce two more abstract levels: case level, and document level
- The new measures build upon the character level ones

Terminology (simplified)

- $s$ denotes a plagiarism case; $S$ a set of plagiarism cases
- $r$ denotes a plagiarism detection; $R$ a set of plagiarism detections
- They refer to two text passages (suspicous and source) in two documents
- We say “$r$ detects $s$” iff source passage and suspicious passage overlap
- $|s|$ and $|r|$ denote the sum character lengths of the passages of $s$ and $r$
- $|r \cap s|$ denotes the length of detection if $r$ detects $s$ in characters
Text Alignment
Performance Measures Revisited

- Currently, detection performance is measured at character level
- We introduce two more abstract levels: case level, and document level
- The new measures build upon the character level ones

Terminology (simplified)

- $s$ denotes a plagiarism case; $S$ a set of plagiarism cases
- $r$ denotes a plagiarism detection; $R$ a set of plagiarism detections
- They refer to two text passages (suspicious and source) in two documents
- We say “$r$ detects $s$” iff source passage and suspicious passage overlap
- $|s|$ and $|r|$ denote the sum character lengths of the passages of $s$ and $r$
- $|r \cap s|$ denotes the length of detection if $r$ detects $s$ in characters

We measure character level precision and recall as follows (simplified):

\[
\text{prec}_{\text{char}}(S, R) = \frac{1}{|R|} \sum_{r \in R} \frac{\sum_{s \in S} |s \cap r|}{|r|}, \quad \text{rec}_{\text{char}}(S, R) = \frac{1}{|S|} \sum_{s \in S} \frac{\sum_{r \in R} |s \cap r|}{|s|},
\]
Text Alignment
Performance Measures Revisited: Case Level

We measure character level precision and recall as follows (simplified):

\[
\text{prec}_{\text{char}}(S, R) = \frac{1}{|R|} \sum_{r \in R} \frac{\sum_{s \in S} |s \cap r|}{|r|}, \quad \text{rec}_{\text{char}}(S, R) = \frac{1}{|S|} \sum_{s \in S} \frac{\sum_{r \in R} |s \cap r|}{|s|},
\]
We measure character level precision and recall as follows (simplified):

\[
\text{prec}_{\text{char}}(S, r) = \frac{\sum_{s \in S} |s \cap r|}{|r|}, \quad \text{rec}_{\text{char}}(s, R) = \frac{\sum_{r \in R} |s \cap r|}{|s|},
\]

where \( S \) and \( R \) represent sets of sentences in the source and target languages, respectively.
Text Alignment
Performance Measures Revisited: Case Level

We measure character level precision and recall as follows (simplified):

\[
\text{prec}_{\text{char}}(S, r) = \frac{\sum_{s \in S} |s \cap r|}{|r|}, \quad \text{rec}_{\text{char}}(s, R) = \frac{\sum_{r \in R} |s \cap r|}{|s|},
\]

Based on these formulas, we define subsets of $S$ and $R$:

- $S' = \{ s \mid s \in S \text{ and } \text{rec}_{\text{char}}(s, R) > \tau_1 \text{ and } \exists r \in R: r \text{ detects } s \text{ and } \text{prec}_{\text{char}}(S, r) > \tau_2 \}$,
- $R' = \{ r \mid r \in R \text{ and } \text{prec}_{\text{char}}(S, r) > \tau_2 \text{ and } \exists s \in S: r \text{ detects } s \text{ and } \text{rec}_{\text{char}}(s, R) > \tau_1 \}$,

where $\tau_1$ and $\tau_2$ determine the least desired character level detection quality:

- $\tau_1, \tau_2 \to 1$ require perfect detection quality
- $\tau_1, \tau_2 \to 0$ allow for poor detection quality
- $\tau_1$ and $\tau_2$ should be set to the top perceptible detection quality
Text Alignment

Performance Measures Revisited: Case Level

We measure character level precision and recall as follows (simplified):

\[
\text{prec}_{\text{char}}(S, r) = \frac{\sum_{s \in S} |s \cap r|}{|r|},
\]

\[
\text{rec}_{\text{char}}(s, R) = \frac{\sum_{r \in R} |s \cap r|}{|s|},
\]

Based on these formulas, we define subsets of \( S \) and \( R \):

\[
S' = \{ s \mid s \in S \text{ and } \text{rec}_{\text{char}}(s, R) > \tau_1 \text{ and } \exists r \in R: r \text{ detects } s \text{ and } \text{prec}_{\text{char}}(S, r) > \tau_2 \},
\]

\[
R' = \{ r \mid r \in R \text{ and } \text{prec}_{\text{char}}(S, r) > \tau_2 \text{ and } \exists s \in S: r \text{ detects } s \text{ and } \text{rec}_{\text{char}}(s, R) > \tau_1 \},
\]

where \( \tau_1 \) and \( \tau_2 \) determine the least desired character level detection quality:

- \( \tau_1, \tau_2 \rightarrow 1 \) require perfect detection quality
- \( \tau_1, \tau_2 \rightarrow 0 \) allow for poor detection quality
- \( \tau_1 \) and \( \tau_2 \) should be set to the top perceptible detection quality

We measure case level precision and recall as follows:

\[
\text{prec}_{\text{case}}(S, R) = \frac{|R'|}{|R|},
\]

\[
\text{rec}_{\text{case}}(S, R) = \frac{|S'|}{|S|}
\]
Text Alignment
Performance Measures Revisited: Document Level

- $D_{plg}$ denotes the set of suspicious documents
- $D_{src}$ denotes the set of source documents
- $D_{pairs} = D_{plg} \times D_{src}$
Text Alignment
Performance Measures Revisited: Document Level

- $D_{plg}$ denotes the set of suspicious documents
- $D_{src}$ denotes the set of source documents
- $D_{pairs} = D_{plg} \times D_{src}$

We define subsets of $D_{pairs}$ based on $S$ and $R$:

$$D_{pairs|S} = \{ (d_{plg}, d_{src}) \mid (d_{plg}, d_{src}) \in D_{pairs} \text{ and } \exists s \in S : d_{plg} \in s \text{ and } d_{src} \in s \}$$

$$D_{pairs|R} = \{ (d_{plg}, d_{src}) \mid (d_{plg}, d_{src}) \in D_{pairs} \text{ and } \exists r \in R : d_{plg} \in r \text{ and } d_{src} \in r \}$$

Likewise, $D_{pairs|R'}$ is based on $R'$ instead of $R$. 
Text Alignment
Performance Measures Revisited: Document Level

- $D_{plg}$ denotes the set of suspicious documents
- $D_{src}$ denotes the set of source documents
- $D_{pairs} = D_{plg} \times D_{src}$

We define subsets of $D_{pairs}$ based on $S$ and $R$:

$$D_{pairs\mid S} = \{(d_{plg}, d_{src}) \mid (d_{plg}, d_{src}) \in D_{pairs} \text{ and } \exists s \in S : d_{plg} \in s \text{ and } d_{src} \in s\}$$

$$D_{pairs\mid R} = \{(d_{plg}, d_{src}) \mid (d_{plg}, d_{src}) \in D_{pairs} \text{ and } \exists r \in R : d_{plg} \in r \text{ and } d_{src} \in r\}$$

Likewise, $D_{pairs\mid R'}$ is based on $R'$ instead of $R$.

We measure document level precision and recall as follows:

$$\text{prec}_{doc}(S, R) = \frac{|D_{pairs\mid S} \cap D_{pairs\mid R'}|}{|D_{pairs\mid R}|}$$

$$\text{rec}_{doc}(S, R) = \frac{|D_{pairs\mid S} \cap D_{pairs\mid R'}|}{|D_{pairs\mid S}|}.$$
Text Alignment
Performance Measures Revisited: Evaluation Results

<table>
<thead>
<tr>
<th>Software Submission</th>
<th>Level of Abstraction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Plagdet</td>
</tr>
<tr>
<td>Team</td>
<td>Year</td>
</tr>
<tr>
<td>Sanchez-Perez</td>
<td>2014</td>
</tr>
<tr>
<td>Oberreuter</td>
<td>2014</td>
</tr>
<tr>
<td>Palkovskii</td>
<td>2014</td>
</tr>
<tr>
<td>Glinos</td>
<td>2014</td>
</tr>
<tr>
<td>Kong</td>
<td>2012</td>
</tr>
<tr>
<td>Shrestha</td>
<td>2014</td>
</tr>
<tr>
<td>Gross</td>
<td>2014</td>
</tr>
<tr>
<td>Oberreuter</td>
<td>2012</td>
</tr>
<tr>
<td>R. Torrejón</td>
<td>2014</td>
</tr>
<tr>
<td>R. Torrejón</td>
<td>2013</td>
</tr>
<tr>
<td>Kong</td>
<td>2013</td>
</tr>
<tr>
<td>Kong</td>
<td>2014</td>
</tr>
<tr>
<td>Palkovskii</td>
<td>2012</td>
</tr>
</tbody>
</table>

- $\tau_1 = 0.5$, so that 50% of a plagiarism case $s$ must be detected
- $\tau_2 = 0.5$, so that 50% of a plagiarism detection $r$ must be a true detection
- Some differences in ranking can be observed
- However, it is still unclear which settings for $\tau_1$ and $\tau_2$ are to be preferred
Summary

PAN 2014

- Source retrieval approaches outperform those of last year
- Twice as much test data as last year
- Too much focus on saving downloads than on saving queries
- Obfuscation prediction to diversify alignment approaches
- Less rule-based extension approaches
- New performance measures

PAN 2015 and beyond

- New text alignment corpora in progress
- New version of ChatNoir based on Elastic Search
- More TIRA support
Summary

PAN 2014

- Source retrieval approaches outperform those of last year
- Twice as much test data as last year
- Too much focus on saving downloads than on saving queries
- Obfuscation prediction to diversify alignment approaches
- Less rule-based extension approaches
- New performance measures

PAN 2015 and beyond

- New text alignment corpora in progress
- New version of ChatNoir based on Elastic Search
- More TIRA support

Thank you for your attention, and your contributions to PAN!