External and Intrinsic Plagiarism Detection using a Cross-Lingual Retrieval and Segmentation System

Markus Muhr, Roman Kern, Mario Zechner, Michael Granitzer
{mmuhr, rkern, mzechner,mgrani}@know-center.at

CLEF 2010 / PAN / 2010-09-22
Overview

Hybrid System

- **External**
  - Based on information retrieval techniques
  - Post-processing based on sequence analysis

- **Intrinsic**
  - Detect style change

- **Cross-lingual plagiarism detection**

- **No heuristics for high obfuscation**
  - No word reordering
  - No synonym resolution

Focus

- Simulate a production system
- Scalable architecture
**System Overview**

**Flowchart**

1. **Source Documents**
   - For each source document:
     - **English?**
       - **No** → **Translate Words**
         - **Segment into overlapping blocks** → **Add blocks to index**
       - **Yes** → **Translate Words**

2. **Suspicious Document**
   - **External**
     - **Segment into small overlapping blocks** → **Use block terms as queries & apply heuristics for fast retrieval** → **Search block index**
       - **Blocks found?**
         - **Yes** → **Similarity & heuristics filtering** → **Blocks** → **Detected passages**
         - **No** → **If no external passages are detected** → **Detected passages**
       - **No** → **Intrinsic**
         - **Segment document into coherent segments** → **Filtering on stylometric features** → **Blocks** → **Detected passages**
   - **Intrinsic**
     - **Token based sequence matching** → **Merge neighboring sequences** → **Blocks** → **Detected passages**
Overview

▶ Two step approach
  ▶ Search for potentially matching suspicious document blocks
  ▶ Apply heuristic post-processing on the potential matches

Work-Flow

▶ Build index out of source documents
  ▶ Build overlapping blocks (40 terms)
▶ Split suspicious documents into blocks (16 terms)
  ▶ Transform blocks into queries
  ▶ Search source index for matching source blocks
Query Construction

▶ For each block in the suspicious document build a query
▶ Sort query terms by document frequency
▶ Join the low frequent terms by AND
▶ Join the remaining terms by OR
▶ Additional heuristics to keep number of queries low
External Plagiarism Detection

Post-Processing

- Starting with query-block pairs
  - Expand the text around the query and the block
  - Build token by token matrix
  - Match for 3 consecutive tokens (and at least 10 characters) - other thresholds for translated documents

- Process the sequences
  - Merged by a neighborhood criterion
  - Finally a similarity between merged sequences is calculated
Cross-lingual Plagiarism Detection

Overview

- Approach: Normalize all documents to English
- Multiple alternative translations
  - Not the single-best translation, but multiple candidates
- Word translations
  - First step of a complete statistical machine translation system
Cross-lingual Plagiarism Detection

Word translations

- Sentence aligned multi-lingual corpus
  - Europarl v5 Koehn [2005]
- Apply word alignment algorithm
  - BerkeleyAligner Liang et al. [2006]
- Number of translation candidates sorted by probability
- Replace each non-English word by up to 5 translation candidates

<table>
<thead>
<tr>
<th>task</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td>no translation</td>
<td>7 ms</td>
</tr>
<tr>
<td>translation</td>
<td>9.38 ms</td>
</tr>
</tbody>
</table>
Intrinsic Plagiarism Detection

Overview

- Style change detection
- Focus on features without semantics

Work-Flow

- Identify regions within a document
- Build feature centroid vector
- Compare regions with centroid
Intrinsic Plagiarism Detection

Region Detection

- First idea: Split document in blocks of equal size
- Approach: Linear text-segmentation algorithm
  - Build blocks of coherent topics
  - Stop-word filtered stems as features
- TextSegFault Kern and Granitzer [2009]
  - Efficient $O(n)$
  - Open-source
Candidate Retrieval Step

- How many false positives are retrieved by the block candidate selection?
- Left: Based on 500 suspicious document in the development corpus
- Right: Based on the evaluation corpus

<table>
<thead>
<tr>
<th>task</th>
<th>hit</th>
<th>all</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>high</td>
<td>2543</td>
<td>3676</td>
<td>0.6918</td>
</tr>
<tr>
<td>low</td>
<td>6614</td>
<td>6988</td>
<td>0.9465</td>
</tr>
<tr>
<td>none</td>
<td>9381</td>
<td>9592</td>
<td>0.9780</td>
</tr>
<tr>
<td>translated</td>
<td>2349</td>
<td>2543</td>
<td>0.9237</td>
</tr>
<tr>
<td>task</td>
<td>hit</td>
<td>all</td>
<td>ratio</td>
</tr>
<tr>
<td>high</td>
<td>13348</td>
<td>14756</td>
<td>0.9046</td>
</tr>
<tr>
<td>low</td>
<td>14832</td>
<td>14883</td>
<td>0.9966</td>
</tr>
<tr>
<td>none</td>
<td>16784</td>
<td>16784</td>
<td>1.0</td>
</tr>
<tr>
<td>translated</td>
<td>5462</td>
<td>6314</td>
<td>0.8651</td>
</tr>
</tbody>
</table>
## Overall System Performance

- Performance results of detected plagiarism separated by different sub-tasks for the hybrid evaluation corpus

<table>
<thead>
<tr>
<th>task</th>
<th>Precision</th>
<th>Recall</th>
<th>Granularity</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-translated all</td>
<td>0.9299</td>
<td>0.8967</td>
<td>1.0553</td>
<td>0.8785</td>
</tr>
<tr>
<td>non-translated none</td>
<td>-</td>
<td>0.9497</td>
<td>1.0025</td>
<td>-</td>
</tr>
<tr>
<td>non-translated low</td>
<td>-</td>
<td>0.9207</td>
<td>1.0968</td>
<td>-</td>
</tr>
<tr>
<td>non-translated high</td>
<td>-</td>
<td>0.8122</td>
<td>1.0771</td>
<td>-</td>
</tr>
<tr>
<td>translated</td>
<td>0.8036</td>
<td>0.61616</td>
<td>2.1655</td>
<td>0.4195</td>
</tr>
<tr>
<td>external</td>
<td>0.9053</td>
<td>0.8631</td>
<td>1.1611</td>
<td>0.7949</td>
</tr>
<tr>
<td>intrinsic</td>
<td>0.212</td>
<td>0.1566</td>
<td>1.0</td>
<td>0.1802</td>
</tr>
<tr>
<td>Overall</td>
<td>0.8417</td>
<td>0.7057</td>
<td>1.1508</td>
<td>0.6948</td>
</tr>
</tbody>
</table>
Conclusions

- Hybrid system
  - External plagiarism detection
  - Support for cross-lingual plagiarism detection
  - Intrinsic (style-based) plagiarism detection

- Issues
  - Scalable (but slow implementation)

- Outlook
  - We plan to build a web service initialized with the Wikipedia as source
The End

Thank you!
