Text Alignment Module in CoReMo 2.1 Plagiarism Detector

Diego A. Rodríguez-Torrejón\textsuperscript{1,2}
José Manuel Martín-Ramos\textsuperscript{1}

\textsuperscript{1} Universidad de Huelva
jmmartin@dti.uhu.es

\textsuperscript{2} I.E.S. José Caballero
dartsystems@gmail.com

The attendance of Diego A. Rodríguez is Penalized by Junta de Andalucía Educational Administration :-(

http://coremodetector.com
Index

- Introduction
- Model Used in Tests
- Context Influence & Surrounding Context N-grams
- Tests Framework
- Test Results
- Conclusions
Comparison from PAN Analysis since '10 to '12 editions shows the mainly common limits to any competitor proposals:

- **Short plagiarism cases** (more frequent into PAN-PC-11) are hardest to detect.

- The former effect is more accused when **crosslingual** cases happen.

- Simulated, low and high paraphrasing cases are much more difficult to detect.
Introduction

Hardest cases uses methods as words removal / replacement / inclusion, sentence reordering, similar appearance character changes...

N-gram based plagiarism detection methods are the most commonly used.

Synonym normalization by WordNet got best results in PAN'11, but it's not enough.

... We need new ways to solve the hardest obfuscation conditions...
Index

- Introduction
- **Model Used in Tests**
- Context Influence & Surrounding Context N-grams
- Tests Framework
- Test Results
- Conclusions
Model Used in Tests
Crosslingual CoReMo

CoReMo System has competed since PAN'10 to PAN'13 achieving the current best Plagdet performance.

The most significant features are the high speed detection and no external translation system dependence, both ideal for intensive tests.

For our first tests, we used our own External PDS: Crosslingual CoReMo 1.7, improved by new Surrounding Context N-grams (SCnG) method. However, SCnG are extensible to any N-gram based PDS (and other IR / NLP tasks).
Model Used in Tests
Crosslingual CoReMo

CoReMo Basics:

- **Extended Contextual N-grams** ($xCTnG$)
- **HAIRS** High Accuracy Inf. Retrieval System only based on n-grams \textit{idf} for local corpora.
- **Reference Monotony Pruning** (RMP)
- **Self-Adaptive Alignment** parameters settings
- **Fast Local Translation** dictionary based
- **External Translation possibility** by scripting
- **Speed Optimized** C/C++ parallel programming
Model Used in Tests
Crosslingual CoReMo

Contextual N-grams* (CTnG) a way to get wide recall and lower index size in sentence order changed environment (translations, active to passive forms …) got by:

- **Case Folding** characters normalization
- **Stopwords** and short length words removal
- **Stemming** by Porter's Stemmer Algorithm
- **N-grams Inner Sort** (after stems selection*)

* **Extended mode** includes stems skipping
Humans can guess a word by near context. In 1977 [16] determined the easiest way: using surrounding context words (a group former and just later).

Usual n-grams belong to closed near context.

**Surrounding Context N-grams (SCnG)** were new concept in '2012 extending CTnG by including new others made from words surrounding a discarded word.

This year **OddEven N-grams (OEnG)** are also included in the model: skip n-grams obtained from odd-only or even-only stems.
Let's see the classic text example (starts from *quick*):

“The *quick brown fox jumps over the lazy dog*”

To get **direct** type xCT3G (CT3G):

1_2_3 → quick brown fox → brown_fox_quick

**Left-hand and Right-hand Context** types (SC3G):

1_2_4 → quick brown jump → brown_jump_quick

1_3_4 → quick fox jump → fox_jump_quick

**Odd n-gram** type (OEnG):

1_3_5 → quick fox laz → laz_fox_quick
Context Influence and Extended Contextual N-Grams

All these n-grams are indexed or compared together. No matter if matching different xCT3G types. This way gets **4 times more n-grams than words** from the same document, increasing the matching opportunities, but **most selectively** than using CT2G: acting as a magnifier effect for the matching context.

Let's see matching possibilities when changes happen:

A) **Changed** word by synonym or any other cause:
   
   “The **quick dark fox** is **jumping** where the dog is”

B) **Text enriching** with new word:

   “The **quick dark brown foxy** jumps where the dog is”
C) Deleted words (summary):

“The brown one jumps over the dog”

D) Translation Errors, writing faults, incorrect term disambiguation: will match as in A case.

The biggest matching quantity enables lowest chunk length to tackle shortest plagiarism cases, without granularity sacrifice or using thesaurus.

xCT3G will get almost the “good” matching opportunities of CT2G, and almost the exceptional precision of CT3G, but improved reliability by its biggest amount, almost without chance noisy matches.
Table 1. n-gram frequency study on PAN-PC-2011 only english source documents subcorpus

<table>
<thead>
<tr>
<th></th>
<th>quantity</th>
<th>ratio</th>
<th>quantity</th>
<th>ratio</th>
<th>quantity</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>idf</td>
<td></td>
<td></td>
<td>CT3G only</td>
<td></td>
<td>CT3G + SC3G</td>
<td></td>
</tr>
<tr>
<td>--</td>
<td>144426869</td>
<td>1.0000</td>
<td>408447501</td>
<td>1.0000</td>
<td>537613396</td>
<td>1.0000</td>
</tr>
<tr>
<td>01</td>
<td>132790997</td>
<td>0.9194</td>
<td>367321473</td>
<td>0.8993</td>
<td>481407991</td>
<td>0.8955</td>
</tr>
<tr>
<td>02</td>
<td>7559052</td>
<td>0.0523</td>
<td>25496723</td>
<td>0.0624</td>
<td>34537949</td>
<td>0.0642</td>
</tr>
<tr>
<td>03</td>
<td>1977892</td>
<td>0.0137</td>
<td>7253659</td>
<td>0.0178</td>
<td>9974359</td>
<td>0.0186</td>
</tr>
<tr>
<td>04</td>
<td>811445</td>
<td>0.0056</td>
<td>3120363</td>
<td>0.0076</td>
<td>4327470</td>
<td>0.0080</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>97</td>
<td>43</td>
<td>0.0000</td>
<td>215</td>
<td>0.0000</td>
<td>265</td>
<td>0.0000</td>
</tr>
<tr>
<td>98</td>
<td>32</td>
<td>0.0000</td>
<td>184</td>
<td>0.0000</td>
<td>260</td>
<td>0.0000</td>
</tr>
<tr>
<td>99</td>
<td>45</td>
<td>0.0000</td>
<td>179</td>
<td>0.0000</td>
<td>261</td>
<td>0.0000</td>
</tr>
<tr>
<td>&gt; 99</td>
<td>1663</td>
<td>0.0000</td>
<td>6379</td>
<td>0.0000</td>
<td>8626</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

About 12,000 docs (1.5 Gbytes plain text)
HAIRS is based in Inverse Document Frequency CTnG study. The best results are got by CT3G
Model Used in Tests
Crosslingual CoReMo

Reference Monotony Prune strategy: 
**discard matching if not happening monotonously.**

Used in several steps to get the fastest runtime, by discarding noisy matching, reducing documents pairs, or complete document comparison even.

- i.e.: Suspicious documents are divided in equal N-gram length chunks. **HAIRS** will get one only document for every chunk.

|    | 73 | -1 | 6 | 49 | 11 | -1 | 31 | 91 | 91 | 91 | 91 | 91 | 6 | 92 | 5 | 7 | 98 | 91 | -1 | -1 |
Plagdet / chunk length

CoReMo 1.6 version only

PAN-PC-2011

monolingual analysis only

- SC3N+Filtro Gr.
- SC3N
- CT3N+Filtro Gr.
- CT3N
Plagdet / chunk length

CoReMo 1.6 version only

PAN-PC-2011

Translated cases only

- SC3G+Filtro Gr.
- SC3G
- CT3G+Filtro Gr.
- CT3G
Every document is modelled having two $xCTnG$ reference lists: naturally ordered and alfabetically ordered ones.

```
FastlyComparableDocument

<vector> NaturalVector : TraceableNgram
<vector> OrderedVector : TraceableNgram
wordLengthAverage : long

setMatchingTo(in otherDocument : FastlyComparableDocument) : void
getDetectionInfo() : string
```
Text Alignment

- When internall order is arranged, internal matching is registered for each xCTnG as a references list.
- The document’s matching cases are got from the ordered lists by a merge-sort modified algorithm, interchanging the references information when matching happens.

<table>
<thead>
<tr>
<th>TraceableNgram</th>
</tr>
</thead>
<tbody>
<tr>
<td>ngram : string</td>
</tr>
<tr>
<td>offset : long</td>
</tr>
<tr>
<td>length : long</td>
</tr>
<tr>
<td>&lt;&lt;list&gt;&gt; innerMatching : TraceableNgram</td>
</tr>
<tr>
<td>&lt;&lt;list&gt;&gt; foreignMatching : TraceableNgram</td>
</tr>
<tr>
<td>compareTo(otherTraceableNgram : TraceableNgram) : int</td>
</tr>
</tbody>
</table>
Reliable matching are those with foreign dtf = 1 and positionally closed to another reliable one in both suspicious and source documents.

When the distance from last reliable match is over the chunk length, the fragment detection finishes, but only will be registered if it's larger than a chunk between the first and the last matches.

The direct detections (seeds) are good, but a bit fragmented. The granularity filter process will join overlapped or closed detections in both documents. We used “only” 4,000 characters distance for this step.

Distances are taken in n-grams for suspicious fragments and in characters for source ones.
These **distances** are got from the **chunk-length** parameter, and also **combined** with word **length average** obtained from the source document.

In order to optimize the tuning for the best performance in the most difficult plagiarism types (summarized) avoiding false positives when no plagiarism cases happens, the **chunk length** \( (cl) \) to different regions depends of the foreign matching rate \( (emr) \) for both documents:

**base case**: \( cl = 8 \times \text{multiplicity factor} \) (4)

- \( emr1 > 4\% \) & \( emr2 < 15\% \) → \( cl = 3 \) \( cl / 7 \)
- \( emr1 > 30\% \) & \( emr2 \geq 15\% \) → \( cl = 2 \) \( cl / 3 \)
## Test Results

### PAN-PC-2013 Training Corpus

<table>
<thead>
<tr>
<th>Obfuscation Type</th>
<th>Plagdet</th>
<th>Recall</th>
<th>Precision</th>
<th>Granularity</th>
</tr>
</thead>
<tbody>
<tr>
<td>No obfuscation</td>
<td>0.92733</td>
<td>0.97326</td>
<td>0.88554</td>
<td>1.00000</td>
</tr>
<tr>
<td>Random obfus.</td>
<td>0.75527</td>
<td>0.63388</td>
<td>0.93417</td>
<td>1.00000</td>
</tr>
<tr>
<td>Translated obfus.</td>
<td>0.84683</td>
<td>0.79951</td>
<td>0.90001</td>
<td>1.00000</td>
</tr>
<tr>
<td>Summary obfus.</td>
<td>0.35513</td>
<td>0.22973</td>
<td>0.87716</td>
<td>1.03529</td>
</tr>
<tr>
<td>Global</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Global bug fixed³</td>
<td>0.82722</td>
<td>0.76758</td>
<td>0.89929</td>
<td>1.00169</td>
</tr>
</tbody>
</table>

### PAN-PC-2013 Competition Corpus

<table>
<thead>
<tr>
<th>Obfuscation Type</th>
<th>Plagdet</th>
<th>Recall</th>
<th>Precision</th>
<th>Granularity</th>
<th>runtime (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No obfuscation</td>
<td>0.92586</td>
<td>0.95256</td>
<td>0.90060</td>
<td>1.00000</td>
<td></td>
</tr>
<tr>
<td>Random obfus.</td>
<td>0.74711</td>
<td>0.63370</td>
<td>0.90996</td>
<td>1.00000</td>
<td></td>
</tr>
<tr>
<td>Translated obfus.</td>
<td>0.85113</td>
<td>0.81124</td>
<td>0.89514</td>
<td>1.00000</td>
<td></td>
</tr>
<tr>
<td>Summary obfus.</td>
<td>0.34131</td>
<td>0.21593</td>
<td>0.90750</td>
<td>1.07742</td>
<td></td>
</tr>
<tr>
<td>Global</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>72508</td>
</tr>
<tr>
<td>Global bug fixed³</td>
<td>0.82220</td>
<td>0.76190</td>
<td>0.89484</td>
<td>1.00141</td>
<td>79965</td>
</tr>
<tr>
<td>Global bug fixed³</td>
<td>0.82827</td>
<td>0.77177</td>
<td>0.89564</td>
<td>1.00140</td>
<td></td>
</tr>
</tbody>
</table>
Test Results

- Most significant improvement are due to SCnG
- Including OEnG and self-tuning improves seeds for precision and Recall, enabling shorter GF.
- Granularity Filter distance is now 1/20th than '12
- A late corrected bug, achieves a even best score:
  \[ \text{PlagDet, Recall, Precision, Granularity, Runtime} \]
  \[ 0.82827 \quad 0.77177 \quad 0.89564 \quad 1.00140 \quad 79965ms \]
- Single core VMs Runtime don't shows real analysis power: CoReMo is now multicore optimized, and we can get same analysis in only 4,5 seconds using 8 cores AMD FX8120 / 4GHz + SSD drive.
Conclusions

- **xCTnG** gets improved detection when harder obfuscation or crosslingual conditions, getting also lower length plagiarism detection.

- **xCTnG** mode gets hoped **CT2G Recall** and practical **CT3G Precision. More and Most Reliable matching Seeds.**

- **Defragmentation** filter gets improved scores at lower detection chunk length. Must be used cautiously however.

- **xCTnG** possibilities open to other IR/NLP tasks.
Future Jobs

- **Improving self-tuning** by studying matching rates distributions, but for chunk length and filter distance also.
- **Improving filtering** by using information of unconnected matches previously discarded.
- **Testing** the possible positive influence of using **Wordnet synsets** reductions, as proposed in PAN'10 and successfully exploded in PAN'11 by J. Grman and R. Ravas.
Acknowledges

• Thanks to the PAN group and all the teams for keeping so interesting challenge every year.

• None entity has supported the Diego Rodríguez job or attendance. It's company (Andalusian Educational Administration) will cut off its salary for the days attending to CLEF2013 : ( 

• To my family, who has enforced me to be here, but its economy (and stability) can not support “Vicious” Research: it has been my ... 

... last-time : ( ???
End ... or Begining?

... But CoReMo will have an opportunity to go on improving only if demonstrates self-financial capability as non-free web services, hoped to start next month and get fully operational about 2014 mid January.

http://www.coremodetector.com
THANKS FOR YOUR ATTENTION

We can improve this slide-show

diego@dartsystems.es
dartsystems@gmail.com
jmmartin@dti.uhu.es
info @ coremodetector.com
References (1)


24. Benno Stein, Paolo Rosso, Efthathos Stammatatos, Moshe Koppel, and Eneko Agirre, editors, SEPLN 09 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN 09), pages 1-9, September 2009. CEUR-WS.org. ISSN 1613-0073
Seeds Comparison

Torrejon13/PAN13 training
Seeds
Plagdet Score 0.77915100343
Recall 0.750258541782
Precision 0.923206830702
Granularity 1.08845070423

Torrejon12/PAN13 training
Seeds
Plagdet Score 0.656719889391
Recall 0.670569425935
Precision 0.922594444295
Granularity 1.26988085342

Torrejon12 / PAN12 Compet. (locally translated)
seeds
Plagdet Score 0.346070995453
Recall 0.419077935863
Precision 0.844858063703
Granularity 2.07139364303

Torrejon13 / PAN12 Compet. (locally translated)
seeds
Plagdet Score 0.408856888467
Recall 0.441193683693
Precision 0.856176743299
Granularity 1.6837565884