Overview of the 2nd International Competition on Plagiarism Detection

Martin Potthast[1], Alberto Barrón-Cedeño[2], Andreas Eiselt[1], Benno Stein[1], and Paolo Rosso[2]

[1]Web Technology & Information Systems
Bauhaus-Universiät Weimar, Germany

[2]Natural Language Engineering Lab, ELiRF
Universidad Politécnica de Valencia, Spain

pan@webis.de http://pan.webis.de

Abstract This paper overviews 18 plagiarism detectors that have been developed and evaluated within PAN'10. We start with a unified retrieval process that summarizes the best practices employed this year. Then, the detectors' performances are evaluated in detail, highlighting several important aspects of plagiarism detection, such as obfuscation, intrinsic vs. external plagiarism, and plagiarism case length. Finally, all results are compared to those of last year's competition.

1 Introduction

Research and development on automatic plagiarism detection is the prominent topic in the broad field of text reuse studies. The evaluation of plagiarism detectors, however, is still in its infancy: recently the first standardized evaluation framework for plagiarism detection has been published [17]. Using this framework the 2nd PAN competition on plagiarism detection was held in conjunction with the 2010 CLEF conference. Altogether 18 groups from all over the world developed plagiarism detectors for PAN, which is 5 more than in last year's competition [16]; 5 groups attended for the second time. In this paper we overview the participants' detection approaches in a comparative manner, and we report on the evaluation of their detection performances.

1.1 Plagiarism Detection

We define a plagiarism case $s = \langle s_{\text{plg}}, d_{\text{plg}}, s_{\text{src}}, d_{\text{src}} \rangle$ as a 4-tuple which consists of a passage s_{plg} in a document d_{plg} that is the plagiarized version of some source passage s_{src} in d_{src}. When given d_{plg}, the task of a plagiarism detector is to detect s, say, by reporting a plagiarism detection $r = \langle r_{\text{plg}}, d_{\text{plg}}, r_{\text{src}}, d'_{\text{src}} \rangle$ which consists of an allegedly plagiarized passage r_{plg} in d_{plg} and its source r_{src} in d'_{src}, and which approximates s as closely as possible. We say that r detects s iff $s_{\text{plg}} \cap r_{\text{plg}} \neq \emptyset$, $s_{\text{src}} \cap r_{\text{src}} \neq \emptyset$, and $d_{\text{src}} = d'_{\text{src}}$. To accomplish this task, the plagiarism detector can resort to two strategies: external plagiarism detection and intrinsic plagiarism detection.

In external plagiarism detection, it is assumed that the source document d_{src} for a given plagiarized document d_{plg} can be found in a document collection D, such as the

Web. Typically, plagiarism detection then divides into three steps [21]: first, a set of candidate source documents D_{src} is retrieved from D, where ideally $|D_{\text{src}}| \ll |D|$ to speed up subsequent computations. Second, each candidate $d_{\text{src}} \in D_{\text{src}}$ is compared in detail with d_{plg}, and a plagiarism detection r is reported if two highly similar passages r_{plg} and r_{src} are identified between d_{plg} and d_{src}. Third, the set R of reported detections is post-processed to filter out false positives.

In intrinsic plagiarism detection, the plagiarism detector attempts to detect plagiarized passages solely based on information extracted from d_{plg} [10]. Strategies for this approach typically include an analysis of d_{plg}'s writing style, since no two authors have the same style. Naturally, detections obtained in this manner do not include source passages and source documents: $r = \langle r_{\text{plg}}, d_{\text{plg}} \rangle$. They are worthwhile nonetheless, since there may be plagiarism cases whose sources have become inaccessible.

1.2 Evaluating Plagiarism Detectors

To evaluate a plagiarism detector we employ our recently published evaluation framework for plagiarism detection, which comprises the PAN plagiarism corpus 2010, PAN-PC-10, and three plagiarism detection performance measures [17]. The plagiarism detector is asked to detect all plagiarism cases in the corpus, and then the accuracy of the detections is measured.

During construction of the PAN-PC-10, a number of different parameters have been varied in order to create a high diversity of plagiarism cases. Table 1 gives an overview: the corpus is divided into documents suspicious of plagiarism, and potential source documents. Note that only a subset of the suspicious documents actually contains plagiarism cases, and that for some cases the sources are unavailable. Also, the fraction of plagiarism per document and the document lengths have been varied. As for the plagiarism cases, one of their most salient properties is whether and how they have been

Table 1. Corpus statistics for 27 073 documents and 68 558 plagiarism cases in the PAN-PC-10.

Document Statistics			
Document Purpose		*Plagiarism per Document*	
source documents	50%	hardly (5%-20%)	45%
suspicious documents		medium (20%-50%)	15%
– with plagiarism	25%	much (50%-80%)	25%
– w/o plagiarism	25%	entirely (>80%)	15%
Detection Task		*Document Length*	
external detection	70%	short (1-10 pp.)	50%
intrinsic detection	30%	medium (10-100 pp.)	35%
		long (100-1000 pp.)	15%

Plagiarism Case Statistics	
Obfuscation	
none	40%
artificial	
– low obfuscation	20%
– high obfuscation	20%
simulated	6%
translated ({de,es} to en)	14%
Case Length	
short (50-150 words)	34%
medium (300-500 words)	33%
long (3000-5000 words)	33%
Topic Match of d_{src} and d_{plg}	
intra-topic cases	50%
inter-topic cases	50%

obfuscated; i.e., real plagiarists rewrite their source passages in order to make detecting them more difficult. A variety of obfuscation strategies have been employed in our corpus, including artificial (automatic) obfuscation, simulated (manual) obfuscation, and automatic translation from German and Spanish to English. Besides the obfuscation type also the length of the plagiarism cases has been varied, and the fact whether or not the topic of a plagiarized document matches that of the source document.

The performance of a plagiarism detector is quantified by the well-known measures precision and recall, supplemented by a third measure called *granularity*. Let S denote the set of plagiarism cases in the suspicious documents of the corpus, and let R denote the set of plagiarism detections the detector reports for these documents. To simplify the notation, a plagiarism case $s = \langle s_{\text{plg}}, d_{\text{plg}}, s_{\text{src}}, d_{\text{src}} \rangle$, $s \in S$, is represented as a set $\mathbf{s}$ of references to the characters of d_{plg} and d_{src} that form the passages s_{plg} and s_{src}. Likewise, a plagiarism detection $r \in R$ is represented as $\mathbf{r}$. Based on these representations, the precision and the recall of R under S can be measured micro-averaged (mic) and macro-averaged (mac):

$$prec_{\text{mic}}(S, R) = \frac{|\bigcup_{(s,r)\in(S\times R)}(\mathbf{s} \sqcap \mathbf{r})|}{|\bigcup_{r\in R} \mathbf{r}|}; \qquad prec_{\text{mac}}(S, R) = \frac{1}{|R|} \sum_{r\in R} \frac{|\bigcup_{s\in S}(\mathbf{s} \sqcap \mathbf{r})|}{|\mathbf{r}|};$$

$$rec_{\text{mic}}(S, R) = \frac{|\bigcup_{(s,r)\in(S\times R)}(\mathbf{s} \sqcap \mathbf{r})|}{|\bigcup_{s\in S} \mathbf{s}|}; \qquad rec_{\text{mac}}(S, R) = \frac{1}{|S|} \sum_{s\in S} \frac{|\bigcup_{r\in R}(\mathbf{s} \sqcap \mathbf{r})|}{|\mathbf{s}|};$$

$$\text{where} \quad \mathbf{s} \sqcap \mathbf{r} = \begin{cases} \mathbf{s} \cap \mathbf{r} & \text{if } r \text{ detects } s, \\ \emptyset & \text{otherwise.} \end{cases}$$

Precision and recall do not account for the fact that plagiarism detectors sometimes report overlapping or multiple detections for a single plagiarism case. This is clearly undesirable, and to address this deficit also a detector's granularity is quantified as follows:

$$gran(S, R) = \frac{1}{|S_R|} \sum_{s\in S_R} |R_s|,$$

where $S_R \subseteq S$ are cases detected by detections in R, and $R_s \subseteq R$ are detections of s. I.e., $S_R = \{s \mid s \in S \wedge \exists r \in R : r \text{ detects } s\}$ and $R_s = \{r \mid r \in R \wedge r \text{ detects } s\}$.

The above measures are computed for each plagiarism detector; however, they do not allow for an absolute ranking among detectors. Therefore, the three measures are combined into a single, overall score as follows:

$$plagdet(S, R) = \frac{F_1}{\log_2(1 + gran(S, R))},$$

where F_1 is the equally-weighted harmonic mean of precision and recall.

2 Survey of Plagiarism Detectors

This section surveys the plagiarism detectors developed for PAN. We summarize the best practices that have been employed this year for external plagiarism detection, dis-

cuss their suitability for practical use, and compare them with those of last year. Finally, Section 2.3 reports on this year's approaches to intrinsic plagiarism detection.

2.1 External Plagiarism Detection in PAN 2010

All except one participant submitted lab reports describing their plagiarism detectors. After analyzing all 17 reports, certain algorithmic patterns became apparent to which many participants followed independently. We have unified and organized the different approaches in the form of a "reference retrieval process for external plagiarism detection": given a suspicious document d and a document collection D, the task is to detect all plagiarism cases s in d. The process follows the aforementioned three steps, i.e., candidate retrieval, detailed analysis, and post-processing.

Candidate Retrieval. In order to simplify the detection of cross-language plagiarism, non-English documents in D are translated to English using machine translation (services). Then, to speed up subsequent computations, a subset D_{src} of D is retrieved that comprises candidates for plagiarism in d. Basically, this is done by comparing d with every document in D using a fingerprint retrieval model: d is represented as a fingerprint $\mathbf{d}$ of hash values of sorted word n-grams extracted from d. Note that sorting the n-grams brings them into a canonical form which cancels out plagiarism obfuscation locally. Beforehand, d is normalized by removing stop words, by replacing every word with a particular word from its synonym set (if possible), and by stemming the remainder. Again, these steps cancel out some obfuscation.

Since many suspicious documents are to be analyzed against D, the entire set D is represented as fingerprint collection, $\mathbf{D}$, which is stored in an inverted index. Then, the postlists for the values in $\mathbf{d}$ are retrieved, and all documents that occur in at least k postlists are considered as candidate source documents D_{src}. Note that the value of k increases as n decreases. This approach is equivalent to an exhaustive comparison of $\mathbf{d}$ with every fingerprint in $\mathbf{D}$ using the Jaccard coefficient, but optimal in terms of runtime efficiency when repeating the task with different suspicious documents.

Detailed Analysis. The suspicious document d is compared in-depth with each candidate source document $d_{\text{src}} \in D_{\text{src}}$. This is done by means of heuristic sequence alignment algorithms, that, inspired by their counterparts in bioinformatics, work as follows: first, the sorted word n-grams that match exactly between $\mathbf{d}$ and $\mathbf{d}_{\text{src}}$ are extracted as seeds. Second, the seeds are merged stepwise into aligned passages by applying merge rules. A merge rule decides whether two seeds or aligned passages can be merged, e.g. by checking whether they fulfill a certain condition with regard to their relative positions in the two documents. Typically, a number of merge rules are organized in a precedence hierarchy: a superordinate rule is applied until no two seeds can be merged anymore by that rule, then the next subordinate rule is applied on the resulting aligned passages, and so on until all rules have been processed. Third, the obtained pairs r_{plg} and r_{src} of aligned passages are returned as plagiarism detections $r = \langle r_{\text{plg}}, d, r_{\text{src}}, d_{\text{src}} \rangle$.

Post-Processing. Before being returned to the user, the set R of plagiarism detections from the previous step is filtered in order to reduce false positive detections. In this regard a set of "semantic" rules is applied that, for instance, require detections to have at least a certain length or that discard detections whose passages r_{plg} and r_{src} do not

exceed a similarity threshold under some retrieval model. Moreover, ambiguous detections that report different sources for approximately the same plagiarized passage in d are dealt with, e.g., by discarding the less probable alternative.

2.2 Discussion and Comparison to PAN 2009

Compared to last year, the detectors have matured and specialized to the problem domain. With regard to their practical use in a real-world setting, however, some developments must be criticized. In general, many of this year's developments pertain only to plagiarism detection in local document collections, but cannot be applied easily for plagiarism detection against the Web.

A novelty this year is that many participants approach cross-language plagiarism cases straightforwardly by automatically translating all non-English documents to English. In cross-language information retrieval, this solution is often mentioned as an alternative to others, but it is hardly ever applied. Reasons for this include the fact that machine translation technologies are difficult to set up in the first place. All of this is of course alleviated to some extent by online translation services like Google Translate.[1] With regard to plagiarism detection, however, this solution can only be applied locally, and not on the Web.

Most of the retrieval models for candidate retrieval employ "brute force" fingerprinting, say, instead of selecting few n-grams from a document, as is custom with near-duplicate detection algorithms like shingling and winnowing [3, 19], all n-grams are used. The average n this year compares to that of last year with about 4.2 words, the winning approach uses 5 words. New this year is that the n-grams are sorted before computing their fingerprint hash values. Moreover, some participants put more effort into text pre-processing, e.g., by performing synonym normalization. Altogether, such and similar heuristics can be seen as *counter-obfuscation* heuristics. Fingerprinting cannot be easily applied when retrieving source candidates from the Web, so some participants employ standard keyword retrieval technologies, such as Lucene and Terrier.[2] All of them, however, first chunk the source documents and index the chunks rather than the documents, so as to retrieve plagiarized portions of a source document more directly. Anyway, be it fingerprinting or keyword retrieval, the use of inverted indexing to speed up candidate retrieval is predominant this year; only few participants still resort to a naïve comparison of all pairs of suspicious documents and source documents.

With regard to detailed analysis, one way or another, all participants employ sequence alignment heuristics, but few notice the connections to bioinformatics and image processing. Hence, due to the lack of a formal framework, participants come up with rather ad hoc rules. Finally, in order to minimize the granularity, some participants discard overlapping detections with ambiguous sources partly or altogether. It may or may not make sense to do so in a competition, but in a real-world setting this cannot hold.

[1] http://translate.google.com

[2] http://lucene.apache.org and http://terrier.org

2.3 Intrinsic Plagiarism Detection in PAN 2010

Intrinsic plagiarism detection has received less attention this year as it turns out that developing algorithms to detect both kinds of plagiarism cases and combining them into a single detector is still too difficult a task to be accomplished within a few months time. Moreover, intrinsic plagiarism detection is still in its infancy compared to external plagiarism detection, and so is research on combining the two. The winning participant reports to have successfully reimplemented last year's best approach, however, the developments were dropped in favor of external plagiarism detection [9]. The third winner has successfully combined intrinsic and external detection by employing the intrinsic detection algorithm only on suspicious documents for which no external plagiarism has been detected [12]. Only one participant has developed an intrinsic-only detector [22]. The underlying approach to intrinsic plagiarism detection has not changed: a suspicious document d is chunked, and, using a writing style retrieval model, each chunk is compared with the whole of d. Then, chunks whose writing style differs significantly from the average writing style of the document are identified using outlier detection. Consecutive outlier chunks are merged, and finally, all outliers are returned as plagiarism detections.

3 Evaluation Results

In this section we report on the detection performances of the plagiarism detectors that took part in PAN. Their overall performance is analyzed, which determines this year's winners, and then, each detector's performance is analyzed with regard to the aforementioned parameters of our evaluation corpus. Again, we discuss the results and compare them with those of last year.

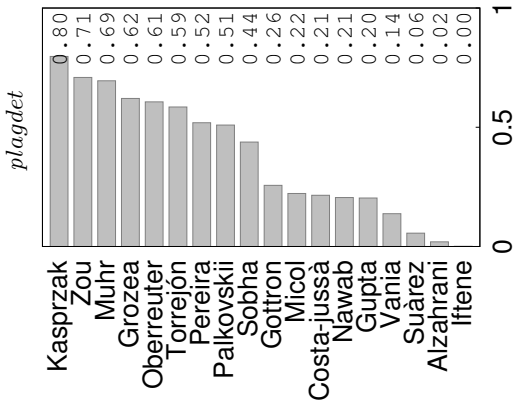
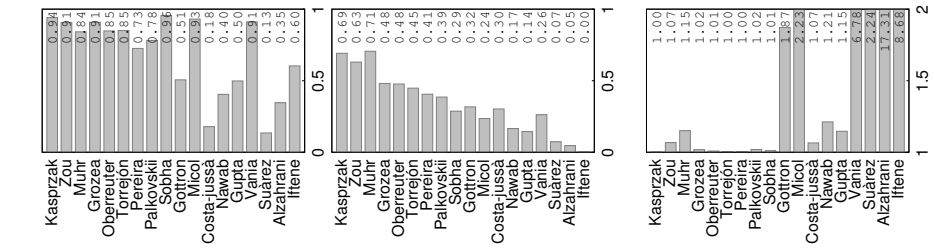


Figure 1. Final ranking of the plagiarism detectors that took part in PAN 2010. For simplicity, each detector is referred to by last name of the lead developer. The plot is best viewed sideways.

Table 2. Plagiarism detection performance on the entire PAN-PC-10.



3.1 Overall Detection Performance

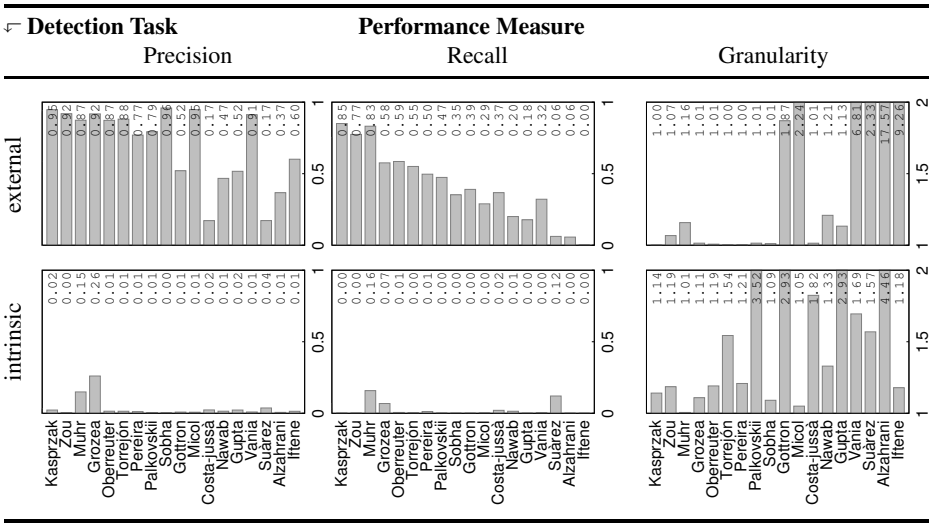
Figure 1 shows the final ranking among this year's 18 plagiarism detectors: each of them was used to detect the plagiarism cases hidden in the PAN-PC-10 corpus, and their overall success is quantified by their $plagdet$ scores. The best performing detector is that of Kasprzak and Brandejs [9], which outperforms both the second and the third detector by about 14%. The remaining detector's performances vary widely from good to poor performance.

In Table 2 each detector's precision, recall, and granularity are given, which were used to compute the $plagdet$ ranking of Figure 1. In every plot the detectors are ordered according to this ranking, so that each performance characteristic can be judged with regard to a detector's final rank. When looking at precision, the detectors roughly divide into two groups: these with a high precision (> 0.7) and these without. Apparently, almost all detectors with a high precision achieve top ranks. The recall is, with some exceptions, proportional to the ranking, while the top 3 detectors are set apart from the rest. Most notably, some detectors achieve a higher recall than their ranking suggests, which pertains particularly to the detector of Muhr et al. [12], which outperforms even the winning detector. With regard to granularity, again, two groups can be distinguished: these with a low granularity (< 1.5) and these without. Remember that a granularity close to 1 is desirable. Again, the detectors with lowest granularity tend to be ranked high, whereas the detectors on rank two and three have a surprisingly high granularity when compared to the others. Altogether, a lack of precision and/or high granularity explains why some detectors with high recall get ranked low (and vice versa), which shows that there is more than one way to excel in plagiarism detection. Nevertheless, the winning detector does well in all respects.

3.2 Detection Performances per Corpus Parameter

Our evaluation corpus comprises a number of parameters that have been varied in order to create a high diversity of plagiarism cases; see Table 1 for an overview. In the following a detailed analysis of the detectors' performances in terms of these parameters is given.

Table 3. Plagiarism detection performance on external / intrinsic plagiarism cases (row 1 / 2).

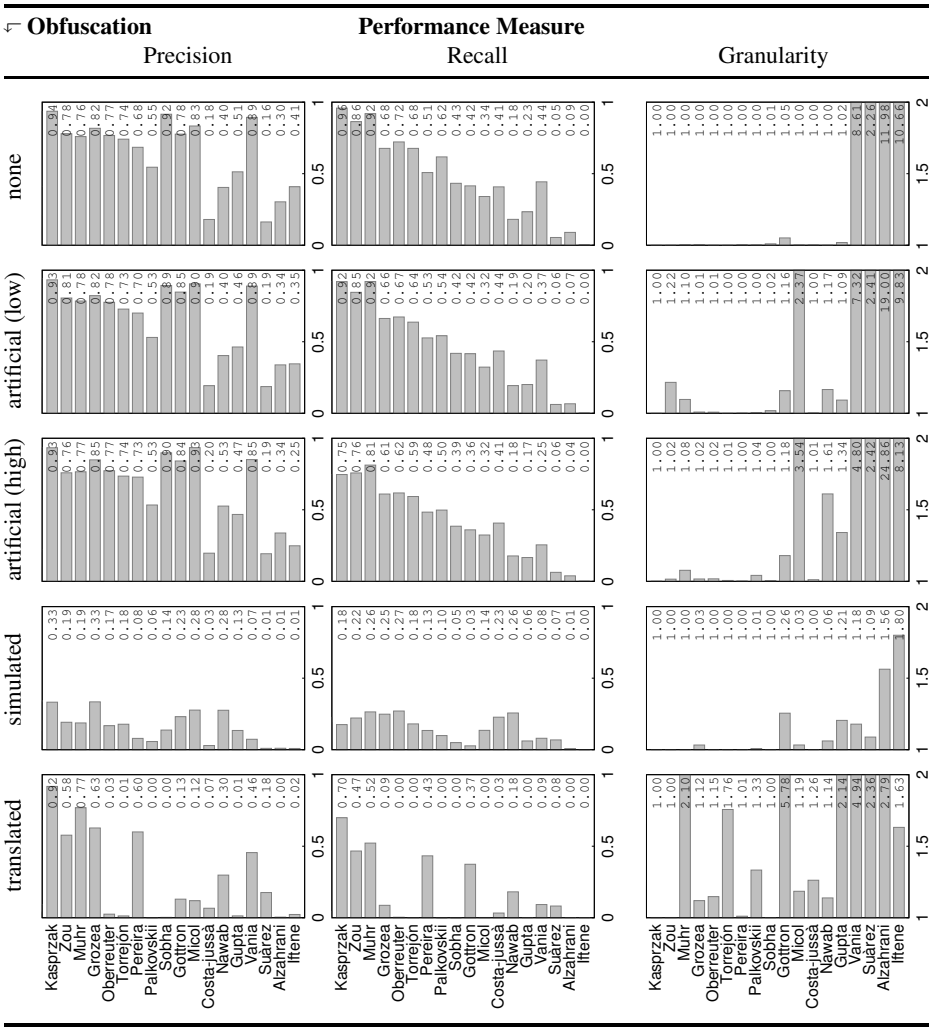


Detection Task. Table 3 summarizes the detection performances with regard to portions of the evaluation corpus that are intended for external plagiarism detection and intrinsic plagiarism detection. Since most of the participants focused on external plagiarism detection, the trends that appear on the entire corpus can be observed here as well. The only difference is that the recall values are between 20% and 30% higher than on the entire corpus, which is due to the fact that about 30% of all plagiarism cases in the corpus are intrinsic plagiarism cases. Only Muhr et al. [12] and Suárez et al. [22] made serious attempts to detect intrinsic plagiarism; their recall is well above 0, but their precision is poor. Nevertheless, combining intrinsic and external detection pays off overall for Muhr et al., and the intrinsic-only detection of Suárez et al. even detects some of the external plagiarism cases. Grozea and Popescu [6] tried to exploit knowledge about the corpus construction process to detect intrinsic plagiarism, which is of course not practical.

Obfuscation. Table 4 summarizes the detection performances with regard to the different obfuscation strategies employed in our corpus. As expected, it is not difficult to detect unobfuscated plagiarism, at least not for the top plagiarism detectors. Artificial plagiarism with both low and high obfuscation can be detected well, too, while the recall decreases slightly with increasing obfuscation. Simulated plagiarism, however, appears to be much more difficult to detect regarding both precision and recall. Interestingly, the best performing detectors on simulated plagiarism are not the top detectors. With regard to translated plagiarism, all participants who used machine translation to first translate non-English documents in the corpus to English were successful both in terms of precision and recall. Some, however, suffer from a poor granularity.

Topic Match. Table 5 summarizes the detection performances with regard to whether or not the topic of a document that contains plagiarism matches that of its source documents. It can be observed that this appears to make no difference at all, other than a slightly smaller precision and recall for inter-topic cases compared to intra-topic cases.

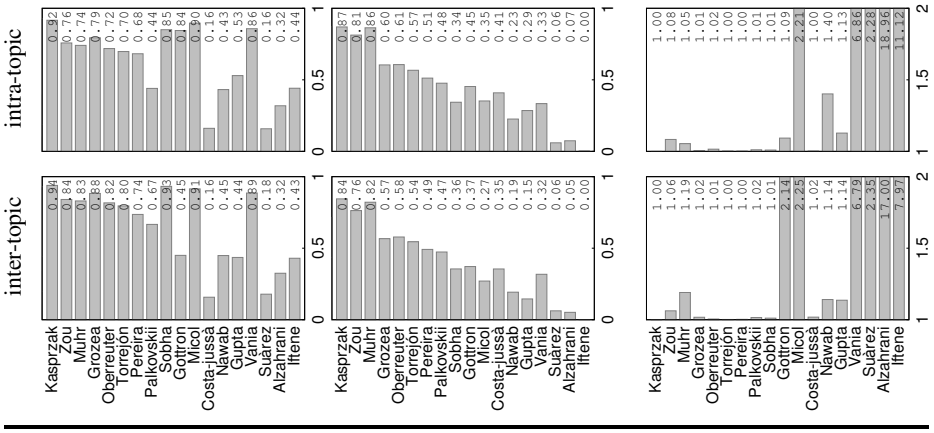
Table 4. Plagiarism detection performance dependent on obfuscation strategy.



However, since many participants did not implement a retrieval process similar to that of Web search engines, some doubts remain whether these results hold in practice.

Case Length, Document Length, and Plagiarism per Document. Tables 6, 7, and 8 summarize the detection performances with regard to the length of a plagiarism case, the length of a plagiarized document, and the percentage of plagiarism per plagiarized document. In general, it can be noted that the longer a case and the longer a document, the easier it is to detect plagiarism. This can be explained by the fact that long plagiarism cases in the corpus are less obfuscated than short ones, assuming that a plagiarist does not spend much time on long cases, and since long documents contain more of the long cases on average than short ones. Also, the more plagiarism per plagiarized document the better the detection, since a plagiarism detector may be more confident with its de-

Table 5. Plagiarism detection performance dependent on whether or not the topic of the plagiarized documents matches that of the source document.



tections if much plagiarism is found in a document. Altogether, the general behavior of the plagiarism detectors with regard to these corpus parameters is similar.

Table 6. Plagiarism detection performance dependent on case length.

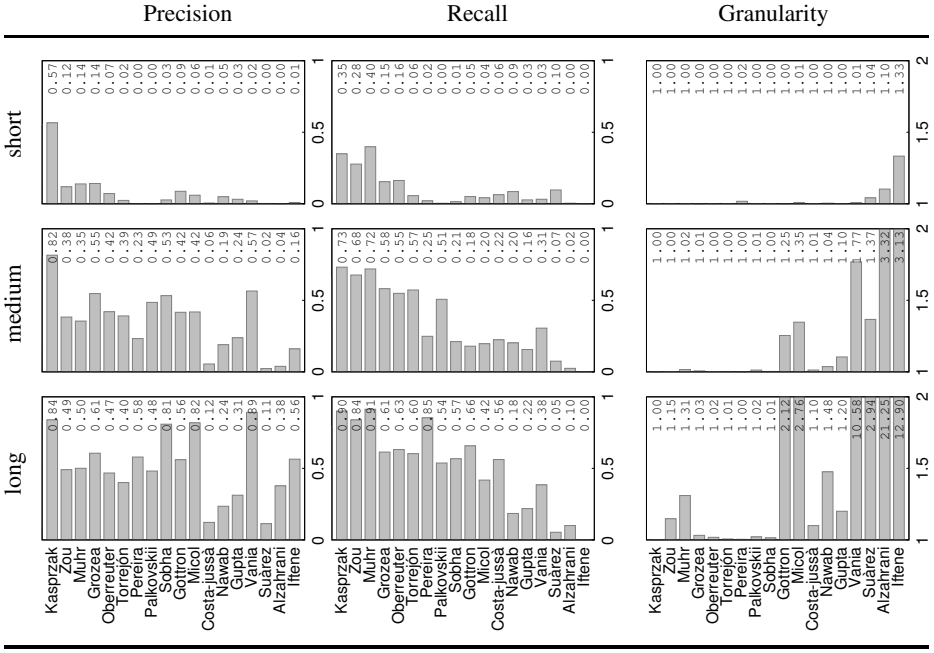
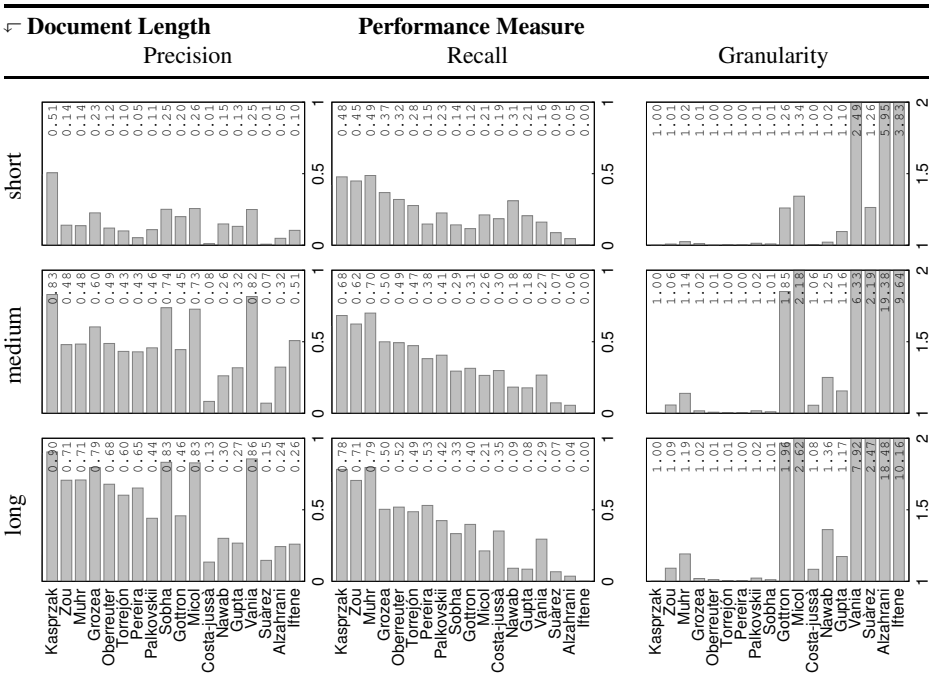


Table 7. Plagiarism detection performance dependent on document length.



3.3 Discussion and Comparison to PAN 2009

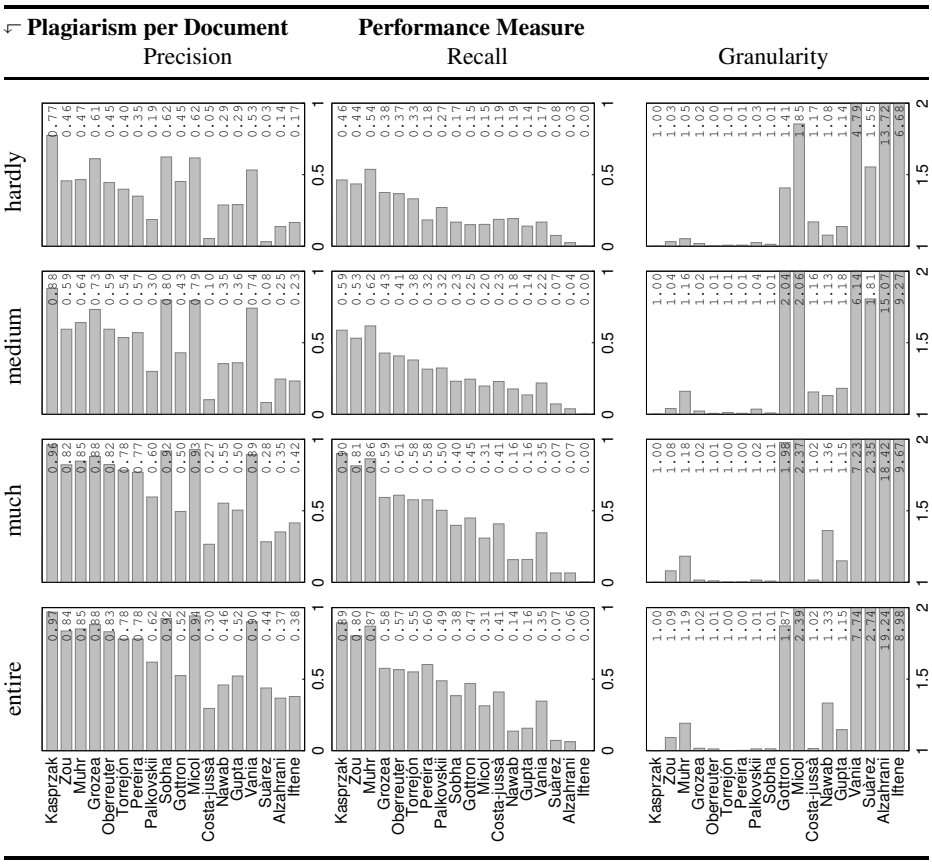
The detection task, the obfuscation strategies, and the length of a plagiarism case are the key parameters of the corpus that determine detection difficulty: the most difficult cases to be detected are those without source, and short ones with simulated obfuscation. The difficulty to detect simulated plagiarism may be in part due to the fact that is was created with specific intructions to create high obfuscation.

Since this year's evaluation corpus has been re-developed from scratch, a comparison to last year's detection results is not straightforward. In this respect, the second-time participation of last year's winner forms an important connection: Grozea and Popescu [6] report that their detector is the same as last year and that almost none of its parameters have been changed. This makes all results of last year comparable to those of this year, simply, by applying the rule of three. Figure 2 shows a combined ranking of all participants from this year and last year. The groups who participated for the second time improved their plagiarism detectors significantly.

4 Conclusion

A number of lessons learned can be derived from this year's results: research and development on external plagiarism detection focuses too much on retrieval from local document collections instead of Web retrieval, while the less developed intrinsic plagiarism detection does not get much attention. Besides Web retrieval, another challenge

Table 8. Plagiarism detection performance dependent on amount of plagiarism per document.



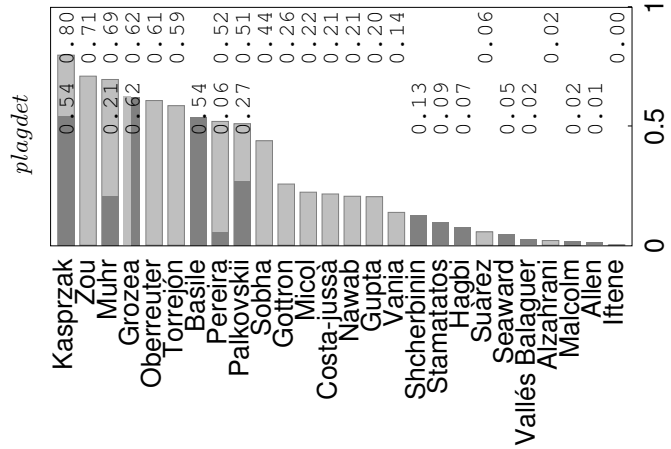


Figure 2. Ranking of the plagiarism detectors that took part in PAN 2009 and PAN 2010. The performances of PAN 2009 detectors are shaded dark, those of PAN 2010 detectors light.

in external plagiarism detection is obfuscation: while artificial obfuscation appears to be detectable relatively easy if a plagiarism case is long, short plagiarism cases as well as simulated obfuscation is not. Regarding translated plagiarism, again, automatically generated cases pose no big challenge in a local document collection, while we hypothesize that simulated cross-language plagiarism will. Future competitions will have to address these shortcomings.

Acknowledgements

We would like to thank our co-organizers Efstathios Stamatatos and Moshe Koppel for their help along the way. Our special thanks go to the participants of the competition for their devoted work. Last not least we thank Yahoo! Research for their sponsorship. This work is partially funded by CONACYT-Mexico and the MICINN project TEXT-ENTERPRISE 2.0 TIN2009-13391-C04-03 (Plan I+D+i).

Bibliography

[1] Salha Alzahrani and Naomie Salim. Fuzzy Semantic-Based String Similarity for Extrinsic Plagiarism Detection: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[2] Martin Braschler, Donna Harman, and Emanuele Pianta, editors. *Notebook Papers of CLEF 2010 LABs and Workshops, 22-23 September, Padua, Italy*, 2010. ISBN 978-88-904810-0-0.

[3] Andrei Z. Broder. Identifying and Filtering Near-Duplicate Documents. In *COM'00: Proceedings of the 11th Annual Symposium on Combinatorial Pattern Matching*, pages 1–10, London, UK, 2000. Springer-Verlag. ISBN 3-540-67633-3.

[4] Marta R. Costa-jussá, Rafael Banchs, Jens Grivolla, and Joan Codina. Plagiarism Detection Using Information Retrieval and Similarity Measures based on Image Processing techniques: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[5] Thomas Gottron. External Plagiarism Detection Based on Standard IR Technology: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[6] Cristian Grozea and Marius Popescu. Encoplot—Performance in the Second International Plagiarism Detection Challenge: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[7] Parth Gupta and Sameer Rao. External Plagiarism Detection: N-Gram Approach using Named Entity Recognizer: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[8] Adrian Iftene. Submission to the 2nd International Competition on Plagiarism Detection, 2010. From the Universtiy of Iasi, Romania.

[9] Jan Kasprzak and Michal Brandejs. Improving the Reliability of the Plagiarism Detection System: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[10] Sven Meyer zu Eißen and Benno Stein. Intrinsic Plagiarism Detection. In Mounia Lalmas, Andy MacFarlane, Stefan Rüger, Anastasios Tombros, Theodora Tsikrika, and Alexei Yavlinsky, editors, *Advances in Information Retrieval: Proceedings of the 28th European Conference on IR Research (ECIR 06)*, volume 3936 LNCS of *Lecture Notes in Computer Science*, pages 565–569, Berlin Heidelberg New York, 2006. Springer. ISBN 3-540-33347-9. doi: http://dx.doi.org/10.1007/11735106_66. URL http://www.springerlink.com/content/x7x483u1k3970863/.

[11] Daniel Micol, Óscar Ferrández, and Rafael Muñoz. A Textual-Based Similarity Approach for Efficient and Scalable External Plagiarism Analysis: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[12] Markus Muhr, Roman Kern, Mario Zechner, and Michael Granitzer. External and Intrinsic Plagiarism Detection using a Cross-Lingual Retrieval and Segmentation System: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[13] Rao Muhammad Adeel Nawab, Mark Stevenson, and Paul Clough. University of Sheffield: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[14] Gabriel Oberreuter, Gaston L'Huillier, Sebastian Rios, and Juan D. Velásquez. FASTDOCODE: Finding Approximated Segments of N-Grams for Document Copy Detection: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[15] Yurii Palkovskii, Alexei Belov, and Irina Muzika. Exploring Fingerprinting as External Plagiarism Detection Method: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[16] Martin Potthast, Benno Stein, Andreas Eiselt, Alberto Barrón-Cedeño, and Paolo Rosso. Overview of the 1st International Competition on Plagiarism Detection. In Benno Stein, Paolo Rosso, Efstathios Stamatatos, Moshe Koppel, and Eneko Agirre, editors, *SEPLN 2009 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN 09)*, pages 1–9. CEUR-WS.org, September 2009. URL http://ceur-ws.org/Vol-502.

[17] Martin Potthast, Benno Stein, Alberot Barrón-Cedeño, and Paolo Rosso. An Evaluation Framework for Plagiarism Detection. In Chu-Ren Huang and Dan Jurafsky, editors, *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 997–1005, Beijing, China, August 2010. Association for Computational Linguistics.

[18] Viviane P. Moreira Rafael C. Pereira and Renata Galante. UFRGSPAN2010: Detecting External Plagiarism: Lab Report for Pan at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[19] Saul Schleimer, Daniel S. Wilkerson, and Alex Aiken. Winnowing: local algorithms for document fingerprinting. In *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 76–85, New York, NY, USA, 2003. ACM Press. ISBN 1-58113-634-X.

[20] Sobha L., Pattabhi R. K Rao, Vijay Sundar Ram, and Akilandeswari A. External Plagiarism Detection: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[21] Benno Stein, Sven Meyer zu Eißen, and Martin Potthast. Strategies for Retrieving Plagiarized Documents. In Charles Clarke, Norbert Fuhr, Noriko Kando, Wessel Kraaij, and Arjen P. de Vries, editors, *30th Annual International ACM SIGIR Conference (SIGIR 07)*, pages 825–826. ACM, July 2007. ISBN 987-1-59593-597-7.

[22] Pablo Suárez, Jose Carlos González, and Julio Villena. A Plagiarism Detector for Intrinsic, External and Internet Plagiarism: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[23] Diego Antonio Rodríguez Torrejón and José Manuel Martín Ramos. CoReMo System (Contextual Reference Monotony) A Fast, Low Cost and High Performance Plagiarism Analyzer System: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[24] Clara Vania and Mirna Adriani. External Plagiarism Detection Using Passage Similarities: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.

[25] Du Zou, Wei-Jiang Long, and Ling Zhang. A Cluster-Based Plagiarism Detection Method: Lab Report for PAN at CLEF 2010. In Braschler et al. [2]. ISBN 978-88-904810-0-0.