# Intrinsic Plagiarism Detection Using Character Trigram Distance Scores

## Notebook for PAN at CLEF 2011

Mike Kestemont, Kim Luyckx, and Walter Daelemans

CLiPS Computational Linguistics Group
University of Antwerp, Belgium
{mike.kestemont, kim.luyckx, walter.daelemans}@ua.ac.be

**Abstract** In this paper, we describe a novel approach to intrinsic plagiarism detection. Each suspicious document is divided into a series of consecutive, potentially overlapping 'windows' of equal size. These are represented by vectors containing the relative frequencies of a predetermined set of high-frequency character trigrams. Subsequently, a distance matrix is set up in which each of the document's windows is compared to each other window. The distance measure used is a symmetric adaptation of the normalized distance ($nd_1$) proposed by Stamatatos [17]. Finally, an algorithm for outlier detection in multivariate data (based on *Principal Components Analysis*) is applied to the distance matrix in order to detect plagiarized sections. In the PAN-PC-2011 competition, this system (second place) achieved a competitive recall (.4279) but only reached a *plagdet* of .1679 due to a disappointing precision (.1075).

*Keywords:* intrinsic plagiarism detection, character n-grams, distance scores, outlier detection, stylometry

## 1 'Intrinsic' plagiarism detection

'Plagiarism' generally refers to the illegitimate use of someone else's information, text, ideas, etc. without proper reference to the original source of these borrowings [9]. A plagiarizing author implicitly claims the original authorship of these borrowings, which in many legal systems is considered a violation of the original author's intellectual property rights. The automatic detection of plagiarized sections in natural language text documents has become a popular task in computational scholarship (e.g. Information Retrieval, Computational Linguistics) [14,12]. The main idea underlying the bulk of this research is that it should be possible to automatically identify plagiarized sections in a text, provided one has access to digital versions of the collection of source documents from which the author might have plagiarized [18]. The availability of an 'external' reference corpus of a suspicious document's potential sources is central to many approaches in automated plagiarism detection. A large number of studies therefore assumes a 'closed world' in which the complete set of works from which a given author might have plagiarized is known beforehand and is readily available in digital form.

Recently, it has been challenged whether this artificial assumption is in fact realistic for real-world cases of plagiarism [11]. On many occasions the potential sources of plagiarisms are not known beforehand or might not even be freely available, let alone searchable in a digital format in the public domain (e.g. world wide web). Moreover, with the increasing amount of easy-access, online information sources nowadays, even a semi-exhaustive search of an author's potential sources becomes increasingly demanding from a computational point of view. In 'intrinsic plagiarism detection', the question is therefore raised whether plagiarized sections can be detected in a suspicious document, *in the absence of any external reference material* [11,10,18]. The idea is that, if an author plagiarized a specific section in a work, one would expect this section to be stylistically deviant from the non-plagiarized sections in the same document that were indeed originally written by the author himself. If this expectation holds true, it should be possible to detect such contaminated sections without having access to an external reference corpus. This kind of plagiarism research bears close similarities to computational authorship studies in the field of stylometry, in which scholars study the correlation between authorial identity and writing style [16,5,7] .

Obviously, the intrinsic variant of plagiarism detection is inherently more difficult than plagiarism detection in studies that depend on external reference material (to a great extent). Note that in its purest formulation, the intrinsic approach does not presuppose that a system has access to external, genuine writings by the suspicious document's main author [18]. The availability of such external training material for supervised learning – a typical experimental set-up in stylometric authorship attribution [8] – would naturally facilitate the task of singling out stylistically deviant passages in an unseen suspicious document. Intrinsic plagiarism detection, however, is not only interesting from a theoretical perspective but is also directly relevant for a number of practical plagiarism scenarios. An interesting example from the world of academia would be a master student hiring a ghost-writer to write one of the chapters of his or her master's thesis. An external plagiarism detector might fail to spot the deviant style of the 'outsourced' chapter, since the ghost writer himself need not have plagiarized. An intrinsic approach, however, might more easily detect the authorial ruptures and alarm the student's supervisor of the compromised integrity of the thesis.

## 2 Document representation

By definition, the intrinsic plagiarism analysis of a suspicious document is limited to an analysis of the suspicious document itself. The initial representation of the suspicious document is therefore vital to an intrinsic approach, determining much of a system's subsequent procedures. The seminal work in this field has been characterized by a fairly standard methodology in this respect (cf. [12]). Typically, a suspicious document is segmented into a series of consecutive (potentially partially overlapping but non-identical) samples or 'windows' of equal size. Window sizes have been typically fixed – variable window sizes have been rarely considered – but the optimal window size is still unclear. On the one hand, stylistic authorship analyses typically require relatively large samples of text (at least ca. 2500 words) while, on the other hand, plagiarized sections need not

be all that long (e.g. a single sentence) [18]. It is therefore common to use segmentation parameters that offer a trade-off between the issues of granularity and performance. Subsequently, it is typical for many studies to compile a sort of feature vector ('document profile') on the basis of the suspicious document as a whole [17]. Next, each (shorter) document window is compared to the (larger) document profile using some sort of distance measure. Finally, stylistically deviant windows are identified using an outlier detection algorithm.

In this paper, we experiment with a novel methodology that departs from this standard document representation and the associated window vs. document profile comparisons. The latter procedure seems troublesome on a number of levels. First of all, from the point of view of stylometry as well as linguistic theory, it seems strange that the relatively smaller windows are compared to the larger profile of the overall document. The frequency distributions of words as well as other style markers are known to be affected by a text's length [1]. Therefore the stylometric comparison of two samples of so different a size (the single window vs. the entire document) is hard to justify from a theoretical perspective. Naturally, distance measures can be used to normalize the effect on the difference in document length (e.g. cosine distance) but even then, this problem seems hard to overcome (cf. [17]). Moreover, the underlying assumption of this approach is that the majority of the suspicious document was genuinely written by a single author. Only in this case, the plagiarized sections would be easily distinguishable from the overall document's profile as outliers. If a disproportionate share (e.g. more than half) of the suspicious document was in fact plagiarized (possibly from various source documents) it seems unlikely that the incoherent profile of such a document would constitute a reliable touchstone for stylistic outlier detection.

In designing our approach, we set out from the hypothesis that comparing a single window to another single window might provide a more reliable methodology than comparing a single window to a much larger entity. With regard to the base sampling parameters, our system nevertheless has the same segmentation parameters (expressed in absolute character counts) as previous approaches: a 'window size' (*ws*) or the length of each window and a 'step size' ($ws >= ss > 0$) determining the number of characters in between the starting points of two consecutive windows. Note that the step size should be larger than zero (to actually proceed through the document while segmenting it) and is preferably smaller than or equal to the window size (in order not to skip any text). The *n* windows that result from this segmentation procedure are then used to create a covariance matrix or distance table with *n* x *n* dimensions. For a document that has been segmented into *n* equal-sized windows $w_1$, $w_2$, ..., $w_{n-1}$, $w_n$ this particular representation is illustrated in Table 1. The cells in this document table are subsequently filled out with distance scores between windows described in the next section.

Windows are represented in terms of character n-grams, as these have been proven useful for style-based categorization (e.g. authorship attribution) [3,6]. They are also able to reliably handle limited data, which is an asset considering the variable length of plagiarized sections (between 50 and 5000 consecutive words).

**Table 1.** Symmetric distance matrix used for the representation of a suspicious document

| | $w_1$ | $w_2$ | ... | $w_{n-1}$ | $w_n$ |
|---|---|---|---|---|---|
| $w_1$ | 0 | $\Delta(w_1,w_2)$ | ... | $\Delta(w_1,w_{n-1})$ | $\Delta(w_1,w_n)$ |
| $w_2$ | $\Delta(w_1,w_2)$ | 0 | ... | $\Delta(w_2,w_{n-1})$ | $\Delta(w_2,w_n)$ |
| ... | ... | ... | ... | ... | ... |
| $w_{n-1}$ | $\Delta(w_{n-1},w_1)$ | $\Delta(w_{n-1},w_2)$ | ... | 0 | $\Delta(w_{n-1},w_n)$ |
| $w_n$ | $\Delta(w_n,w_1)$ | $\Delta(w_n,w_2)$ | ... | $\Delta(w_n,w_{n-1})$ | 0 |

## 3 Distance score

The distance score we have adopted is an adapted version of the normalized distance ($nd_1$) proposed by Stamatatos [17]. This distance operates on the level of character *n*-grams, whereby a text is divided into a series of overlapping character groups of length *n*. Under a third order character *n*-gram model (*n*=3), the word 'plagiarism' would for instance include the following character trigrams (with whitespace represented as an underscore): {'_pl', 'pla', 'lag', 'agi', 'gia', 'iar', 'ari', 'ris', 'ism', 'sm_'}. When calculating the original $nd_1$ between the windows $w_x$ and $w_y$, a list is created of all *n*-grams found in $w_x$ (but not necessarily in $w_y$). This collection is called the 'profile' of $P(w_x)$ with $|P(w_x)|$ denoting the window's absolute length. This profile is used to calculate the normalized distance between two windows using the following formula, where $f_{w_x}(g)$ represents the frequency of trigram g in $w_x$:

$$nd_1(w_x, w_y) = \sum_{g \in P(w_x)} \frac{\left(\frac{2(f_{w_x}(g) - f_{w_y}(g))}{f_{w_x}(g) + f_{w_y}(g)}\right)^2}{4|P(w_x)|}$$

The denominator ensures that the real number resulting from this dissimilarity function will lie between 0 (extreme similarity) and 1 (extreme dissimilarity). Note that this approach is computationally expensive when dealing with large text collections, since each comparison is based on a different set of *n*-grams that for each comparison will have to re-established. Moreover, this measure is not symmetric: because in calculating $nd_1(w_x,w_{y\neq x})$ only the trigrams encountered in $P(w_x)$ will be considered so that $nd_1(w_x, w_{y\neq x}) \neq nd_1(w_{y\neq x}, w_x)$.

We therefore propose a novel use of $nd_1$ (denoted $\Delta$ here for simplicity), whereby the original formula is not applied to every $g \in P(w_x)$ but, instead, to every *n*-gram in a predefined set of high-frequency *n*-grams that is the same for all suspicious documents used in an experiment. This suggestion is inspired by work on authorship attribution in stylometry which has shown that high-frequency linguistic items (and in particular *n*-grams) show excellent performance in comparison to other, more difficult to extract features [16]. This adaptation can be justified in terms of efficiency as well and efficacy. Note that this adaptation of Stamatatos' normalised distance [17] is symmetric, meaning that $\Delta(w_x,w_{y\neq x}) = \Delta(w_{y\neq x},w_x)$, while $\Delta(w_x,w_x = 0)$ by definition. As such,

each row in e.g. Table 1 can be considered a vector that describes one window's behavior in terms of its distance to all other windows in the same document. Note that such a representation is reminiscent of the kind of distance matrices used in statistical clustering.

## 4 Outlier detection

Based on the distance matrix representing a suspicious document, one can now try to detect stylistically deviant sections. Given our particular text representation, this task becomes similar to outlier identification in multivariate data sets. In case of long documents and small window sizes, our representation can be high dimensional. In our system, we have therefore used a technique for outlier identification in high dimensions, proposed by Filzmoser, Maronna and Werner [4] as implemented in the *mvoutlier* package for the R statistical software package [15]. This technique has been optimized for data sets in which the number of dimensions is very high or even severely outnumbers the number of observations (note that in our case both are equal). The technique is furthermore computationally efficient because it reduces the size of the data set by applying a *Principal Components Analysis* (PCA), a standard technique for dimensionality reduction, commonly applied in stylometry [2]. Subsequently, a robust version of the Mahalanobis distance (commonly used for outlier identification) is used to detect outliers in the most informative dimensions resulting from the PCA. Interestingly, the algorithm assigns different weights to different components, because outliers will tend to be extremely clear in one component, while relatively absent in others. The software will eventually output a boolean decision for each observation, indicating which windows can be considered stylistic 'outliers' and thus potentially plagiarized. In our system the characters of all windows returned as 'outliers' by the *mvoutlier* package were assumed to be plagiarized. Adjacent and overlapping windows, however, were concatenated into a single plagiarism instance in order to ensure a good granularity of our approach.

## 5 Experimental Results and Discussion

In this section, we will report some of our experimental results on the training and test corpora used in the PAN-PC-2011 competition. All alphabetic characters in the suspicious documents were lowercased, but apart from this, no other preprocessing steps (e.g. reduction of multiple whitespaces) were taken. This decision ensured close comparability with the character indices used to denote plagiarized fragments. Our distance metric is dependent on a set of high-frequency character *n*-grams: we extracted a list of all *n*-grams from the entire suspicious document's corpus from the 2010 competition and ranked them according to their cumulative, absolute frequency. For each experiment, we selected the *n* most frequent *n*-grams from the top of this list (e.g. *n*=1,000). Due to a lack of time, we only experimented with character trigrams and mainly focused on the effect of the segmentation parameters on the document representation.

Table 2 presents the results of a series of experiments, exploring the effect of the segmentation parameters *ws* and *ss* on the overall performance of the system on the

**Table 2.** Experimental results on training corpus for PAN-PC-2011 for *outbound*=.25 and *n*=1,000. Exploration of the effect of segmentation parameters *ws* and *ss*.

| window size | step size | plagdet | recall | precision | granularity |
|---|---|---|---|---|---|
| 20,000 | 20,000 | 19.48 | 20.02 | 19.01 | 1.00 |
| 20,000 | 15,000 | 20.59 | 21.84 | 19.88 | 1.01 |
| 20,000 | 10,000 | 23.80 | 27.79 | 21.00 | 1.01 |
| 20,000 | 5,000 | 25.84 | 39.55 | 19.52 | 1.02 |
| 20,000 | 1,000 | 26.36 | 44.99 | 18.91 | 1.01 |
| 15,000 | 15,000 | 20.04 | 20.29 | 20.71 | 1.01 |
| 15,000 | 11,250 | 22.41 | 23.09 | 22.41 | 1.02 |
| 15,000 | 7,500 | 25.97 | 29.69 | 23.44 | 1.01 |
| 15,000 | 3,750 | 26.79 | 40.17 | 20.63 | 1.02 |
| 15,000 | 750 | **27.21** | 45.09 | 19.89 | 1.02 |
| 10,000 | 10,000 | 21.33 | 20.35 | 23.34 | 1.03 |
| 10,000 | 7,500 | 24.14 | 24.05 | 25.95 | 1.05 |
| 10,000 | 5,000 | 27.26 | 29.98 | 25.89 | 1.03 |
| 10,000 | 2,500 | **27.53** | 40.00 | 22.03 | 1.04 |
| 5,000 | 5,000 | 21.77 | 20.38 | 28.09 | 1.12 |
| 5,000 | 3,750 | 24.03 | 24.18 | 29.79 | 1.16 |
| 5,000 | 2,500 | **27.52** | 30.42 | 28.50 | 1.10 |
| 5,000 | 1,250 | **27.49** | 37.56 | 24.55 | 1.11 |

training corpus for the PAN-PC-2011 competition. We report on the figures concerning precision, recall, granularity as well as the overall plagdet score [13]. The figures were calculated using the reference implementation available from the competition's website. From this table it is clear that the system can more easily reach a higher recall (max. 45.09) than a higher precision (max. 29.79). Smaller window sizes seem to yield better scores and the same seems true for smaller step sizes (e.g. $ss <= ws/2$), while the granularity is only slightly worse with these settings. Because of the large difference between precision and recall, we have subsequently tried to tune the *outbound* parameter of the *pcout* function, 'a numeric value between 0 and 1 indicating the outlier boundary for defining values as final outliers (default to 0.25)' [4]. Table 3 covers a number of fairly random experiments that show that a lower outbound value tended to boost the recall even more, while higher outbound values slightly pushed the system's precision (cf. Table 2).

We submitted a test run for the competition with the following settings $ws = 5,000, ss = 2,500, n = 2,500, outbound = .20$, which on the training corpus reached a *plagdet* of 28.60, a recall of 36.37, a precision of 26.7 and a granularity of 1.11. We chose these parameters, because the average document length in the test corpus was smaller than in the training corpus and we were speculating that these settings (e.g. higher *n*) were better suited to handle the fine-grained analysis of such shorter documents. An *outbound* of .20 was selected to ensure high recall. The result of these settings on the test corpus were a plagdet of 16.79, a recall of 42.79, a precision of 10.75 and a granularity of 1.03, resulting in a second place. Whereas we find a more

**Table 3.** Experimental results on training corpus for PAN-PC-2011 for n=1,000. Exploration of the effect of the outbound-parameter.

| outbound | window size | step size | plagdet | recall | precision | granularity |
|---|---|---|---|---|---|---|
| .20 | 20,000 | 20,000 | 19.92 | 21.17 | 18.84 | 1.00 |
| .20 | 20,000 | 5,000 | 25.87 | 41.84 | 19.06 | 1.02 |
| .30 | 20,000 | 5,000 | 25.66 | 36.60 | 20.09 | 1.01 |
| .30 | 15,000 | 3,750 | 26.82 | 37.24 | 21.48 | 1.02 |
| .35 | 15,000 | 3,750 | 25.68 | 30.01 | 22.91 | 1.02 |
| .30 | 10,000 | 2,500 | **27.61** | 36.93 | 23.13 | 1.04 |
| .20 | 10,000 | 2,500 | 27.29 | 42.25 | 21.17 | 1.04 |

than competitive recall, precision for the test corpus is surprisingly low, as compared to results from the development phase.

The system identified 18,691 cases of plagiarism (with low precision), where only 11,443 needed to be detected. While a lot of short (< 1,000 characters) plagiarized passages were not detected, keeping window size and step size relatively high did result in reasonable scores for longer passages. A rough error analysis shows that the system did not detect any plagiarism in the majority of cases with manually obfuscated text. A smaller step size might increase the system's precision, but would be more computationally expensive.

## 6 Conclusions

In this paper, we introduced a novel type of document representation for the intrinsic plagiarism detection task. While the standard methodology – comparing a profile of the full document to every smaller window in the text [12,17] – assumes the majority of the suspicious document was written by a single author, our approach is not hindered by this premise. By comparing a single window to all equal-sized windows from the document and applying outlier detection, we can detect stylistic outliers. In order to keep computational cost within bounds, we rely on a *predetermined* set of high-frequency character trigrams and consequently apply a symmetric adaptation of the normalized distance ($nd_1$) proposed by Stamatatos [17].

During the development phase, we experimented with a number of variables, such as window size, step size, and the *outbound* parameter of the outlier detection algorithm. Although our specific selection of parameters returned high recall and reasonable precision, the actual test run scored competitively in recall but disappointed in terms of precision. Short and medium-length plagiarized sections seem to be particularly challenging for our approach.

## Acknowledgements

Mike Kestemont is a Ph.D fellow of the Research Foundation – Flanders (FWO). The research of Luyckx and Daelemans is partially funded through the IWT project AMiCA: Automatic Monitoring for Cyberspace Applications.

## References


1. Baayen, R.H.: Word Frequency Distributions, Text, Speech and Language Technology, vol. 18. Kluwer (2001)
2. Binongo, J.N., Smith, W.: The application of principal components analysis to stylometry. Literary and Linguistic Computing 14(4), 445–466 (1999)
3. Clement, R., Sharp, D.: Ngram and Bayesian classification of documents for topic and authorship. Literary and Linguistic Computing 18(4), 423–447 (2003)
4. Filzmoser, P., Maronna, R., Werner, M.: Outlier identification in high dimensions. Computational Statistics and Data Analysis 52(3), 1694–1711 (2008)
5. Juola, P.: Authorship attribution. Found. Trends Inf. Retr. 1, 233–334 (December 2006), http://portal.acm.org/citation.cfm?id=1373450.1373451
6. Keselj, V., Peng, F., Cercone, N., Thomas, C.: N-gram-based author profiles for authorship attribution. In: Proceedings of the 6th Conference of the Pacific Association for Computational Linguistics. pp. 255–264. Pacific Association for Computational Linguistics, Halifax, Canada (2003)
7. Koppel, M., Schler, J., Argamon, S.: Computational methods in authorship attribution. J. Am. Soc. Inf. Sci. Technol. 60, 9–26 (January 2009), http://portal.acm.org/citation.cfm?id=1484611.1484627
8. Luyckx, K., Daelemans, W.: The effect of author set size and data size in authorship attribution. Literary and Linguistic Computing 26(1), 35–55 (2011)
9. Maurer, H., Kappe, F., Zaka, B.: Plagiarism - A Survey. Journal of Universal Computer Science 12(8), 1050–1084 (2006)
10. Meyer zu Eißen, S., Stein, B.: Intrinsic Plagiarism Detection. In: Lalmas, M., MacFarlane, A., Rüger, S., Tombros, A., Tsikrika, T., Yavlinsky, A. (eds.) Advances in Information Retrieval: Proceedings of the 28th European Conference on IR Research (ECIR 06). Lecture Notes in Computer Science, vol. 3936 LNCS, pp. 565–569. Springer, Berlin Heidelberg New York (2006), http://www.springerlink.com/content/x7x483u1k3970863/
11. Meyer zu Eißen, S., Stein, B., Kulig, M.: Plagiarism Detection without Reference Collections. In: Decker, R., Lenz, H.J. (eds.) Advances in Data Analysis. Selected Papers from the 30th Annual Conference of the German Classification Society (GfKl). pp. 359–366. Springer (2007)
12. Potthast, M., Barrón-Cedeño, A., Eiselt, A., Stein, B., Rosso, P.: Overview of the 2nd International Competition on Plagiarism Detection. In: Notebook Papers of CLEF 2010 LABs and Workshops (2010)
13. Potthast, M., Stein, B., Barrón-Cedeño, A., Rosso, P.: An Evaluation Framework for Plagiarism Detection. In: Proceedings of the 23rd International Conference on Computational Linguistics, COLING 2010 (2010)
14. Potthast, M., Stein, B., Eiselt, A., Barrón-Cedeño, A., Rosso, P.: Overview of the 1st International Competition on Plagiarism Detection. In: Proceedings of the 3rd Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse (2009)
15. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2011), http://www.R-project.org, ISBN 3-900051-07-0
16. Stamatatos, E.: A Survey of Modern Authorship Attribution Methods. Journal of the American Society for Information Science and Technology 60(3), 538–556 (2009)
17. Stamatatos, E.: Intrinsic Plagiarism Detection Using Character N-gram Profiles. In: Proceedings of the 3rd International Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (2009)
18. Stein, B., Lipka, N., Prettenhoffer, P.: Intrinsic Plagiarism Analysis. Natural Language Engineering 45(1), 63–82 (2011)